PROBABILISTIC MODEL-BASED APPROACH TO EVOLUTIONARY ANALYSIS OF NON-CODING SEQUENCES

BY

JAEBUM KIM

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Computer Science
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2010

Urbana, Illinois

Doctoral Committee:

Assistant Professor Saurabh Sinha, Chair
Professor Jiawei Han
Associate Professor ChengXiang Zhai
Assistant Professor Jian Ma

Abstract

Non-coding sequences, constituting a large fraction of genomic DNA, are of great importance because (i) they harbor functional elements that are involved in the regulation of gene expression and (ii) they are essential for the study of genome structure and evolution. The availability of genome sequences of closely related species has provided opportunities to analyze non-coding sequences by comparing multiple genomes from different species. The success of comparative genomic studies relies on bioinformatics tools that aid the comparison and analysis of genome sequences.

Here, we propose and develop computational tools to evolutionarily analyze non-coding sequences, which are based on probabilistic models of sequence evolution. We present a probabilistic framework for finding the locations of insertions and deletions (indels) in a multiple alignment. Its performance is found to be better than that obtained by a parsimony-based method. We study the evolution of sequences involved in the regulation of body patterning in the *Drosophila* embryo, reporting statistical evidence in favor of key evolutionary hypotheses related to regulatory elements and constraints on indels. We also propose a new simulation scheme for generating biologically realistic benchmarks for the alignments of non-coding sequences. This scheme is used to construct benchmarks for *Drosophila* non-coding sequences, and evaluation results are shown for several multiple alignment and indel annotation tools on those benchmarks. Finally, we develop a probabilistic framework for multiple sequence alignment that finds an optimal alignment by incrementally building up alignment columns, based on a model for the evolution of three sequences and the joint probability of an alignment column as a substitute for the traditionally used sum-of-pairs score. We find that the new framework produces alignments of much greater specificity than state-of-the-art methods, without compromising too much in terms of sensitivity.

The computational tools developed here will play a significant role in solving many biological problems and further contribute to broaden our understanding of organismal diversity and

evolution.

To my family.

Acknowledgments

This dissertation would never have been possible without the support of many people. I would sincerely thanks to my advisor, Prof. Saurabh Sinha, for his great guidance for my development as a researcher and invaluable advice throughout my research. I thanks to my committee members, Prof. Gustavo Caetano-Anolles, Prof. Jiawei Han, Prof. Jian Ma, and Prof. ChengXiang Zhai, who offered precious comments and perspectives on my work. I also thanks to Prof. Sang-goo Lee at Seoul National University in Korea for inspiring and motivating me to start shaping my career as a researcher.

I thanks to my colleagues, Charles Blatti, Jia-Yu Chen, Ryan Cunningham, Thyago S. Duque, Xin He, Andra Ivan, Majid Kazemian, Jin Tae Kwak, Xu Ling, and Md Abul Hassan Samee, for giving me valuable comments on my research and making the life in school fun. I especially thanks to Xin He, who helped me to do a great work on one of my projects.

My life in Urbana-Champaign was exciting and fun thanks to my friends, Daniel Ahn, Kanghoon Jun, Jin Huh, SangKyum Kim, Younhee Ko, Yoonkyong Lee, Eun Soo Seo, and others in the computer science department. I thanks to them for sharing my memorable and enjoyable life in Urbana-Champaign.

Personally, I truly thanks to my family, my wife Jung-Ae, daughters Gahyun and Gyuhyun, and parents. Without their endless support and love, I would never have finished my dissertation. Finally, it would be impossible to completely express the steadfast patience and effort from my wife, Jung-Ae. Thank you.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Non-coding sequences are portions of genomic DNA that do not code for proteins. They comprise a large proportion of a genome, e.g., approximately 98% of *Homo sapiens* and 83% of *Drosophila melanogaster* [1]. Non-coding sequences are of great importance because they contain functional elements, many of which are involved in regulating gene expression. Gene expression is a process by which information encoded in genes is converted into proteins or functional RNAs [20]; gene regulation determines when and where a gene is expressed, and to what extent. Gene regulation plays a crucial role in the development and functioning of cells, and the understanding of the mechanisms underlying the evolution of gene regulation provides key insights into the diversity within and between species.

Non-coding sequences are also important for studying patterns of mutations, genome structure and evolution. For example, pseudogenes, which are relatives of genes yet have lost the ability to code for proteins, have been used to study patterns of spontaneous mutation, such as substitutions, insertions and deletions (indels) [64, 126, 127, 185]. The indel spectrum obtained from the analysis of pseudogenes has been used to elucidate the determinants of genome size [125, 129]. Introns are DNA sequence regions within a gene that are transcribed to precursor mRNAs but are not used to code for proteins. Intron length has been found to have a correlation with recombination rate in *Drosophila* [27], and constraints on introns may determine the different patterns of insertions versus deletions in introns versus pseudogenes in *Drosophila* [121, 137]. Conserved regions within both intronic and intergenic sequences have been analyzed to compare the strength of constraints [6, 55] on non-coding sequences. There was also an effort to explain evolution of intron length in *Drosophila* by using the patterns of indels [136].

The availability of genome sequences of closely related species has provided opportunities to solve many key biological problems by comparing these genomes. This research paradigm, called

comparative genomics [57, 108], is based on the premise that common features of two organisms are often encoded within conserved DNA sequence regions. For example, if genes in two related species are regulated similarly, non-coding sequence regions controlling those genes should be relatively conserved. However, DNA sequence regions responsible for differences between different species are not expected to be conserved. The success of comparative genomics relies on bioinformatics tools to investigate and analyze genomes, and the basic requirements are: (i) the alignment program that highlights regions of homology among sequences and predicts nucleotide level relationships among them, and (ii) the annotation program for evolutionary events, such as substitutions and indels, in an alignment. A lot of such bioinformatics tools have been being developed by various researchers. The availability of multiple tools offers us not only flexibility but also difficulty in choosing the most appropriate tool(s) for the bioinformatics analyses. As a result, it is also indispensable to develop benchmarks that help researchers to evaluate and select the most effective tool(s).

The main focus of this dissertation is on developing computational tools to aid the evolutionary analysis of non-coding sequences, which are based on probabilistic models of sequence evolution. We present a probabilistic framework for finding the locations of indels in a multiple alignment (Chapter 3). The framework finds the maximum likelihood annotation of indels. Its performance is found to be better than that obtained by a parsimony-based method. We next analyze regulatory sequence evolution in 12 *Drosophila* species (Chapter 4). Here, we study the evolution of sequences involved in the regulation of body patterning in the *Drosophila* embryo and report statistical evidence in favor of key evolutionary hypotheses, such as characteristics of regulatory elements (transcription factor binding sites (TFBSs)), determinants of the gain and loss of TFBSs in evolution, and constraints on indels. To aid the evaluation and selection of the most appropriate tool(s), we propose a new simulation scheme for generating biologically realistic benchmarks for the alignments of non-coding sequences (Chapter 5). This scheme is used to construct benchmarks for the alignments of *Drosophila* non-coding sequences, and evaluation results are shown for several multiple alignment and indel annotation tools on those benchmarks. Finally, we develop a probabilistic framework for multiple sequence alignment (Chapter 6). The framework is based on a probabilistic model of sequence evolution and aims to find the maximum expected accuracy alignment by incrementally building up alignment columns. By using the benchmarks developed

in Chapter 5, we find the performance of the new framework is overall comparable or superior to existing multiple alignment tools.

Chapter 2

Related Work

2.1 Annotation of Insertions and Deletions in a Multiple Sequence Alignment

Traditional alignment programs mark the predicted locations of insertions and deletions (indels) as gaps and do not proceed to annotate these gaps as being indels. This latter task has received some attention recently and two indel annotation schemes have been proposed: one is based on maximum-parsimony and the other on a probabilistic model.

2.1.1 Parsimony-Based Indel Annotation

The maximum-parsimony method annotates indels in a multiple alignment by minimizing the total number of indel events [50, 158]. Blanchette et al. [10] extended the standard maximum-parsimony method by introducing costs for indels and developed a greedy algorithm for reconstructing ancestral sequences. This method has been successfully applied to the reconstruction of ancestral sequences of mammals [10]. Snir and Pachter [158] developed an efficient parsimony-based indel annotation algorithm and compared indel events between coding and non-coding regions. Chindelevitch et al. [23] provided an efficient inference algorithm for parsimonious indel evolutionary histories based on a linear programming technique. Chen et al. [22] developed a web server for the identification of non-species-specific as well as species-specific indel events based on multiple sequence alignments from at least three species. Tanay and Siggia [167] discovered that sequence context has an effect on the rate of short indels by using a maximum-parsimony annotation, and developed a probabilistic model for predicting the potential of indels at a genomic locus by analyzing its sequence context.

2.1.2 Probabilistic Model-Based Indel Annotation

An alternative to the maximum parsimony approach is to infer the history of indel formation by applying a probabilistic model of sequence evolution. The simplest way is to take advantage of the well studied model of substitution and extend it to incorporate indels. Rivas [140] developed an extended Markov substitution model that includes gaps as the fifth DNA base. However, a problem in this approach is the assumption of alignment column independence even though consecutive gaps are the result of a single mutation event. Nevertheless, applications of the model to the problem of phylogenetic inference were found to lead to a gain in accuracy compared to a method that does not consider gaps [141]. Diallo et al. [35] used a "tree hidden Markov model" (tree-HMM) to reconstruct the most likely scenario of indels. The tree-HMM is a probabilistic model that can represent the process for the evolution within a single alignment column. States of the tree-HMM consist of all possible assignment of indel events to branches in a phylogenetic tree. A similar idea was used to define a probabilistic framework, called "transducers", for modeling indels on a phylogenetic tree [13]. The transducers can represent the evolution of the input sequence into the output sequence based on a probabilistic model of substitutions and indels. Indel annotation can be automated by placing the transducers on each branch of a phylogenetic tree. Paten et al. [123] applied the transducers to develop a tool for genome-wide reconstruction of mammalian ancestral sequences.

2.2 Multiple Sequence Alignment

Multiple sequence alignment (reviewed in [4, 34, 78, 85, 118]) is the task of aligning three or more sequences simultaneously to discover biological and evolutionary relationships among them. Aligned residues are said to be homologous to each other, which means that they have evolved from a common ancestral residue. Multiple sequence alignment is an essential preprocessing step for downstream evolutionary analyses of biological sequences. Despite the importance of a multiple sequence alignment, it is not a trivial task due to computational burden. For example, the extension of a pairwise dynamic programming algorithm [116] based on a hidden Markov model (HMM) to the alignment of multiple sequences requires $O(N_s^2 L^N)$ time and $O(N_s L^N)$ memory, where N is

the number of sequences, N_s is the number of states in the HMM, and L is the average length of sequences. The most popular heuristic to reduce the complexity of multiple sequence alignment while producing reasonable alignments is the so-called progressive alignment method [46]. There are a few variants in use, but the basic procedure of a progressive alignment method is that two sequences are chosen and aligned by a standard pairwise alignment, and next another sequence is chosen and aligned to the alignment that was generated and fixed in the previous step. This process repeats until all sequences are aligned. The most serious drawback of this method is that errors introduced at early stages cannot be resolved at later stages. To address this problem, most progressive alignment methods are coupled with an iterative refinement step. However, the iterative refinement cannot correct all alignment errors, especially complex ones. The problem of the progressive alignment method stems from its large alignment step, in the sense that a whole sequence is aligned to another sequence or alignment at each stage. Many studies have attempted to reduce the step size of the progressive alignment method. For example, a segment-to-segment alignment method used in DIALIGN program [110, 111] progressively aligns multiple sequences by an incremental construction of alignment segments. The sequence annealing method proposed recently by Schwartz and Pachter [148] takes crucial further steps in this direction by allowing for segments of size one.

2.2.1 Scoring a Multiple Sequence Alignment

Traditional score-based multiple alignment programs, such as Mlagan [18], use a fixed substitution scoring matrix and penalties for gaps obtained from limited analyses with a few sets of sequences. To address the problem of the traditional score-based approach, a recent probabilistic model-based alignment method computes the posterior probabilities of aligning two residues in different sequences [37, 124] and uses them to define a measure of similarity to the "true alignment", called "expected accuracy". The parameters controlling the probabilistic model can be easily estimated from given input sequences. The expected accuracy scheme was extended to take advantage of the conservation information from the comparison of multiple sequences [37]. However, the problem of the expected accuracy method is that it is a sensitivity measure in the sense that it does not impose penalties on over-aligned residue pairs. Therefore it may lead to over-alignment where biologically

unrelated residues are aligned. To resolve this problem, a new accuracy measure, called "Alignment Metric Accuracy" (AMA), was developed by Schwartz and Pachter [148] and successfully applied to the FSA multiple alignment tool [14]. In AMA, the posterior probabilities of unaligned residues together with those of aligned ones are used to assess the accuracy of multiple sequence alignments, and this results in a balanced consideration of alignment sensitivity and specificity. In addition, by controlling the contribution of the probabilities of unaligned residues, one can generate multiple alignments with either higher sensitivity or higher specificity.

2.2.2 Model of Sequence Evolution

In the course of evolution, extant sequences have been created by evolutionary processes for residue substitutions and indels. The probabilistic modeling of the evolutionary processes is a natural way to discover evolutionary relationships among given sequences (reviewed in [102]). The pioneering work by Thorne et al. [173] introduced a continuous-time evolutionary model (the TKF91 model) for sequence substitutions and indels, and showed how to calculate the likelihood of two sequences under the model. This model assumes indels to be single-residue events. In order to define a more realistic model, Thorne et al. [174] extended the TKF91 model to handle indels of fragment of residues (the TKF92 model), whose lengths follow a geometric distribution. However, the fragment boundaries are fixed over all ancestral sequences and therefore the TKF92 model does not allow overlapping indels. A more general model, called "long indel" model [107], was later developed. The long indel model removed the assumption of non-overlapping indels of the TKF92 model at the cost of time complexity.

Many studies have attempted to use or extend the above evolutionary models to handle multiple sequences. The first attempt, by Steel and Hein [161], handled sequences related by a star tree. They later extended the method to multiple sequences in an arbitrary phylogenetic tree [61]. A hidden Markov model (HMM) framework provides us many efficient algorithms, such as estimating parameters and finding maximum likelihood interpretation, and the TKF91 model has been successfully described as a pair-HMM [66]. Hein et al. [63] constructed a multiple-HMM based on the TKF91 model and this kind of HMM construction is also applicable to the TKF92 model. Holmes [65] showed how to construct a multiple-HMM by using transducers that represent the

evolution of a sequence on each branch of a phylogenetic tree. A rigoruos probabilistic treatment is computationally intractable because of its high time and memory complexity, and therefore limited to a small number of sequences. There has been an effort to accelerate the probabilistic alignment algorithms by simplifying the original recursion algorithm in the TKF91 model [62, 101]. To reduce the time and space requirements, a heuristic corner-cutting method was successfully applied to multiple alignments [62]. The corner-cutting method only keeps high contributing regions in a dynamic programming table to the maximum likelihood alignment and discards non-contributing ones (typically non-diagonal regions). A sampling-based approach, such as Markov chain Monte Carlo (MCMC) [66] that samples multiple alignments from the posterior probability distribution of alignments, was also developed to make the probabilistic alignment practical to handle large number of sequences.

2.3 Benchmarking Tools for Multiple Alignments

2.3.1 Simulation of Non-Coding Sequence Evolution

Simulation-based benchmarks have been widely used to assess bioinformatics tools for multiple alignments of non-coding sequences. As a result, many simulation programs for the evolution of DNA sequences have been developed. Seq-Gen [139] is a simulation program for DNA sequences that supports three models of nucleotide substitutions: HKY [59], F84 [45], and general reversible process (REV) [184] model. However, indels are not included in the simulation model. The program Rose [163] is based on a probabilistic model of the evolution of RNA, DNA, and protein sequences, and allows for the creation of indels. However, its substitution model is limited to simple models. The program Dawg [21] was developed to address these limitations of previous programs. Dawg can simulate evolution of DNA sequences based on a continuous time reversible substitution model and a novel length-dependent indel model. Recently, more generalized simulation programs were developed [49, 164] and they can support new features, such as motif conservation and flexible indel evolutionary model.

2.3.2 Assessing Multiple Sequence Alignments

Assessing multiple sequence alignments is a difficult task because the "true alignment" is unknown. As an attempt to assess multiple sequence alignments without resorting to the true alignment, Prakash and Tompa [134] developed a tool, called StatSigMA, to check whether a multiple alignment is contaminated with one or more unrelated sequences based on the statistics of local multiple alignments. They later extended the method to handle genome-wide alignments [135]. Landan and Graur [86] showed that a reasonable surrogate for the accuracy of an alignment program on a data set can be computed without the true alignment. They reasoned that good alignments should be invariant to the orientation of the input sequences, and therefore defined the Heads or Tails (HoT) alignment quality score as the agreement between two alignments, one generated from original sequences and the other from their reversed versions. Hall [54] showed that there is a clear positive correlation between HoT alignment quality scores and the real alignment accuracy measured by comparison with the true alignment. Later on, Landan and Graur [87] extended the HoT method to take advantage of co-optimal alternative alignments generated by progressive alignment tools.

2.3.3 Benchmarking Multiple Alignment Tools

Many benchmarking studies have been done to evaluate and compare multiple alignment tools. For example, Thompson et al. [170] performed a comprehensive comparison of multiple alignment tools that use different alignment schemes based on BAliBASE benchmark [171] for protein sequences. To compare the ability of aligning non-coding sequences, Pollard et al. [132] simulated non-coding sequences with four different configurations based on the presence and absence of conserved regions and indels, and evaluated multiple alignment tools. One of the problems of popular progressive alignment is that errors introduced early in the alignment process cannot be corrected. To address this problem, many multiple alignment tools have utilized iterative refinement methods in slightly different implementation. Therefore, Wallace et al. [177] compared multiple iterative alignment algorithms by using protein sequence benchmarks [109]. There was also an effort to discover the limits of pairwise and multiple alignment tools for aligning non-coding sequences with a simulation based benchmarks [133].

Chapter 3

Probabilistic Model-Based Annotation of Insertions and Deletions in a Multiple Alignment

3.1 Background

The recent publication of genome sequences of several closely related species [103] provides a great opportunity to study the molecular events underlying evolution. It is now possible to do large scale comparison of homologous sequences and measure the extent of sequence evolution due to various categories of change, such as substitutions, insertions and deletions, and then look for interesting patterns. For instance, neutral sequences in *Drosophila*, such as non-functional transposons or pseudogenes, show an excess of deletions over insertions [126], while our recent work [154] found evidence for a different pattern, viz., an excess of insertions over deletions, in regulatory sequences. These preliminary findings have raised the question whether there is a clear difference in patterns of evolutionary events between neutral and functional non-coding sequences, either due to mechanistic reasons or due to selection. An answer to this question is of fundamental scientific significance, and an affirmative answer may help pinpoint functionality of certain kinds in non-coding sequence.

The basic requirements for a study of evolutionary events are: (i) an alignment program and (ii) an annotation program to identify the insertions and deletions in the alignment. Existing alignment tools such as Blastz [150] may be assumed to return broad areas of homology ($\sim$1Kbp or longer), that can be accurately aligned by programs like Mlagan [18] and TBA [11]. The annotation program should then identify the substitutions, insertions and deletions so as to explain the gaps in the alignment, and be efficient enough to do this on genomic scales, for modest numbers of species. In this chapter, we describe a new algorithm with this functionality, in a probabilistic framework.

The standard way to quantify evolutionary events currently is a two step approach: (i) obtain a multiple alignment of the sequences, and (ii) annotate the insertions and deletions in the alignment using maximum parsimony criteria [10, 50, 156]. This approach has the following problems,

however:

- The maximum parsimony approach does not afford a principled way to weigh insertions and deletions of various lengths against each other, and against substitutions, when counting events. Its results are expected to be inferior to that from a more general likelihood-based approach that incorporates substitutions, insertions and deletions in a unified evolutionary model.

- The alignment step is completely decoupled from the annotation step. However, if there is a prior expectation about rates of various events, it may be possible to obtain an alignment that is more in tune with the prior expectation, *if alignment and annotation were integrated.* The converse is also possible, i.e., to estimate various evolutionary parameters based on the observed alignment (over large numbers of samples).

- The alignment step is often based on an underlying dynamic programming algorithm to optimize a scoring function that does not explicitly use a model of evolution, and makes no distinction between insertions and deletions.

Here, we propose a new algorithm, called "Indelign", that addresses the above problems in current strategies for recording evolutionary events. Let the term **"evolutionary history"** refer to a multiple alignment and an annotation of insertions/deletions along the branches of the phylogenetic tree, consistent with the gaps in the alignment. Indelign evaluates an evolutionary history by its likelihood of being generated by a model, whose parameters are the rates of substitution, insertion and deletion, and length distributions of insertions and deletions. The Indelign program can be used for any of the following goals, for three or more species, given their phylogeny with relative branch lengths:

- Given a multiple alignment, *annotate* the insertions and deletions on each branch of the phylogeny so as to maximize the likelihood of the resulting evolutionary history.

- Given a multiple alignment that is close to optimal, *make limited changes to the alignment* so that the resulting evolutionary history is consistent with the assumed model parameters.

- Given a set of multiple alignments produced by similar evolutionary processes, learn the values of the model parameters and the most likely evolutionary history simultaneously.

An interesting application of Indelign will be to fit the model parameters on multiple alignment of closely related species, such as the seven sequenced strains of *D. simulans*, where the alignments are accurate, and then apply the trained model to infer evolutionary histories of more diverged species. This application has been proposed earlier in [77]. The accurate annotation of insertions and deletions, which may involve inferring the sequences at the internal nodes of the phylogeny, will also play an important role in the emerging field of ancestral genome reconstructions [10].

We prefer to designate our program as an "indel annotation and realignment program". Its goal is not to search the entire space of multiple alignments; rather to start from a reasonably good alignment, obtained from existing multiple alignment programs, and improve it to get accurate estimates of insertions and deletions. The major thrust of our work is on accurate annotation of insertions and deletions (indels).

Our work has similarities to the MCAlign algorithm of Keightley and Johnson [77], that finds a maximum likelihood alignment of non-coding sequences using an evolutionary model, whose parameters include rates of substitutions and indels, *and the length distribution of indels*. The possibility of multiple indels happening at the same locus is ignored in MCAlign, as in Indelign. However, MCAlign is implemented for at most three species, while Indelign is completely general in terms of number of species, and scales well with the number of species. The separate modeling of insertions and deletions is an important feature of Indelign, that is absent in MCAlign. Moreover, MCAlign assumes that the model parameters are known, e.g., from multiple alignments of more closely related species, while Indelign allows for estimation of the parameters from the input data using an iterative algorithm. We note that MCAlign is an alignment program whereas Indelign is developed primarily as an indel annotation program.

Fredslund et al. [50] and Blanchette et al. [10] have proposed parsimony-based algorithms for annotating insertions and deletions. Indelign, on the other hand, finds the best annotation by maximum likelihood using an evolutionary model that integrates insertions, deletions, and substitutions in a principled manner.

Statistical approaches to phylogenetic inference have a rich history going back to Jukes and

Cantor [73], Kimura [82] and Felsenstein [44], where the process of nucleotide substitution was modeled at various levels of biological realism. Indels were first included in a rigorous probabilistic model by Thorne et al. [173], who defined a continuous-time, time-reversible evolutionary process with single nucleotide indels. Later developments by Holmes and Bruno [66], Metzler [106] and Miklós et al. [107] built on this model, or its extension to longer indels ("TKF92" [174]), to provide statistical alignment algorithms that can allow for accurate inference of evolutionary histories, but these methods are unlikely to scale to genome-wide analysis. Indelign takes a pragmatic approach to the problem, with an explicit goal of summarizing evolutionary event statistics in three or more species, for the restricted but extensively studied domain of closely related genomes, e.g., the mammalian genomes [10] or the fruitfly genomes. Importantly, Indelign, like MCAlign, provides the advantage of allowing arbitrary length distributions of indels (which may optionally be trained from the data). This aspect of the model makes the computational tasks of training the model and evaluating likelihoods challenging, and Indelign implements new ideas for solving the problem efficiently.

Several interesting ideas on evolutionary models that have inspired our work may be found in *simulation* programs developed by Cartwright [21], Stoye et al. [163] and Pollard et al. [132]. These programs incorporate fairly sophisticated evolutionary models, including separate and flexible treatment of insertions and deletions, but differ from our work in that their goal is to *generate* synthetic sequence data related by a phylogeny, rather than to annotate existing alignments with their evolutionary events.

3.2 Model and Likelihood

We begin with an input of (a) a multiple alignment of sequences, each sequence being a string in Σ^L, where $\Sigma = \{A, C, G, T, -\}$ and L is the alignment length, as well as (b) a phylogenetic tree $T = (V, E)$, where $V = V_L \cup V_I$, V_L is the set of leaf nodes, V_I is the set of internal nodes, and E is the set of branches. Let S_v be the sequence at node v, and is known for all $v \in V_L$.

3.2.1 Formalizing Evolutionary Histories

A complete evolutionary history (**CEH**) assigns to each node $v \in V_I$ a sequence $S_v \in \Sigma^L$ (Figure 3.1A). An indel evolutionary history (**IEH**) assigns to each node $v \in V_I$ a sequence $S_v \in \{*, -\}^L$, i.e., it only specifies whether each position of S_v is a gap ($-$) or an unknown nucleotide ($*$) (Figure 3.1A). An IEH can therefore correspond to a large number of CEHs (which may be exponential in the sequence length), obtained by "instantiating" each of the '$*$'s at the internal nodes in one of four ways. An IEH (or CEH) uniquely specifies the insertion, deletion, and alignment (orthology) events on all branches $e \in E$, as per the following rules (Figure 3.1B). ($P(e)$ and $C(e)$ denote the parent and child nodes of edge e respectively.)

- A deletion (D, e, l, r), on branch $e \in E$, located at position l to r, is possible only if (a) the child node sequence has only gaps in those positions (i.e., $S_{C(e)}[l \ldots r]$ is all gaps) and (b) the parent node sequence has non-gaps at positions l and r (i.e., $S_{P(e)}[l] \neq$ '$-$' and $S_{P(e)}[r] \neq$ '$-$').

- An insertion (I, e, l, r), on branch $e \in E$, located at position l to r, is possible only if (a) the parent node sequence has only gaps in those positions (i.e., $S_{P(e)}[l \ldots r]$ is all gaps) and (b) the child node sequence has non-gaps at positions l and r (i.e., $S_{C(e)}[l] \neq$ '$-$' and $S_{C(e)}[r] \neq$ '$-$').

- An alignment (A, e, p) exists at position p on branch e iff $S_{C(e)}[p]$ and $S_{P(e)}[p]$ are both non-gaps. Additionally, for a CEH, if $S_{P(e)}[p] \neq S_{C(e)}[p]$, a substitution is said to have occurred.

- For every branch e, and every position p, either (i) there exists either an alignment (A, e, p) or an indel $(D/I, e, l, r)$ such that $p \in [l, r]$, or (ii) $S_{P(e)} = S_{C(e)} =$ '$-$'.

3.2.2 Evolutionary Model

Overview

The evolutionary model describes the probabilities of various event types (substitution, insertion and deletion) along each edge of the phylogeny. Under the model, every base in the ancestral sequence is prescribed a certain probability of (i) being substituted (or remaining conserved), (ii)

being the start position of a deletion, or (iii) having an insertion occur immediately preceding it. The probabilities of these different event types depend on the branch length (evolutionary time), event type, and various model parameters. Important properties of the model are noted at the end of the following paragraph. For computational tractability, we impose two intuitively justified restrictions on the evolutionary histories that we will consider (and evaluate under the model); these are presented in Section 3.2.3.

The likelihood of a CEH φ is defined recursively as follows (S_v^{φ} is the sequence assigned in φ to node v):

$$L(\varphi) = Pr(S_{root}^{\varphi}) \prod_{e \in E} Pr(S_{P(e)}^{\varphi} \rightarrow S_{C(e)}^{\varphi}) \tag{3.1}$$

For any edge e, the probability $Pr(S_{P(e)} \rightarrow S_{C(e)})$ is prescribed by an evolutionary model that has a hidden Markov model (HMM) structure defined by the parent sequence $S_P(e)$. This model is illustrated in Figure 3.2. $S_{P(e)}$ is first "trimmed" to remove all gaps, and for each position i in the trimmed sequence $S_{P(e)}^{tr}$, we add to the model (i) a "Begin deletion" (BD_i) state, (ii) an "Align" (A_i) state, and (iii) an "Insertion" (I_i) state, with inter-connections as in the illustrative widget in Figure 3.2. The widget for each position is linked to widgets for other positions in a strictly left-to-right manner. (There is an additional "Insertion" state (I_{END}) for the rightmost end of $S_{P(e)}^{tr}$.) The "Align" state A_i emits a nucleotide by applying a suitable substitution process (described below) to the i^{th} nucleotide of the parent sequence $S_{P(e)}^{tr}$. An "Insertion" state emits a sequence with length drawn from the insertion length distribution, and the sequence itself is sampled from the stationary distribution of the substitution process. The "Begin deletion" state has no emission; rather, a positive deletion length is chosen from the deletion length distribution, and as many "positions" in the HMM are skipped by the next transition. The probability of a sequence being generated by this HMM prescribes the evolutionary transition probability $Pr(S_{P(e)} \rightarrow S_{C(e)})$ on branch e. We next note a few important features of the evolutionary model. (i) The length distributions of insertions and deletions are unconstrained parameters of the model. (ii) The indel process is in "discrete time", i.e., an insertion (or deletion) is created with a fixed probability that depends on the time spanned by the entire branch. (iii) As in Dawg [21], there are no restrictions on the relative rates of insertions and deletions, which implies that the model is not time reversible. (iv) Indels do not happen inside other indels on the same branch. This is a realistic assumption for

species not too far diverged, and leads to a greatly simplified and tractable model. Note, however, that the HMM model does allow multiple indels adjacent to each other.

The model prescribes computation of the probability $Pr(S_{P(e)} \rightarrow S_{C(e)})$ (from Equation 3.1) as

$$
\begin{aligned}
Pr(S_{P(e)} \rightarrow S_{C(e)}) &= A_e \times I_e \times D_e \\
A_e &= \prod_{i|S_{P(e)}[i]\neq - \ \cap \ S_{C(e)}[i]\neq -} Pr(S_{P(e)}[i] \rightarrow S_{C(e)}[i]) \\
I_e &= {p_I}^{N_I}(1-p_I)^{\overline{N_I}} \prod_{(I,e,l,r)} Pr_I(r-l+1) \\
D_e &= {p_D}^{N_D}(1-p_D)^{\overline{N_D}} \prod_{(D,e,l,r)} Pr_D(r-l+1)
\end{aligned}
$$

Here, the terms A_e, I_e, D_e on the right hand side correspond to the likelihood contributions from the alignments, insertions and deletions respectively. p_I and p_D are transition probabilities of the "Insertion" and "Begin deletion" states (See Figure 3.2.), N_I and N_D are the number of insertions and deletions, $\overline{N_I}$ and $\overline{N_D}$ are the number of positions where insertions and deletions were not made. $Pr_I(\cdot)$ and $Pr_D(\cdot)$ are the probability distributions on insertion and deletion lengths respectively.

Substitution Model

The term A_e in Equation 3.2 may use any time-dependent single nucleotide substitution model. Our current implementation uses the continuous-time Felsenstein model [44]. (This choice was arbitrary, and it is straight-forward to implement other substitution models if necessary.) This model has a single parameter u which, along with the stationary distribution of nucleotides $\{\pi_A, \pi_C, \pi_G, \pi_T\}$ defines the transition probabilities for any branch of length t (in arbitrary units):

$$Pr(\alpha \rightarrow \beta|t) = (1-e^{-ut})\pi_\beta \quad \texttt{where } \alpha \neq \beta \tag{3.2}$$

$$Pr(\alpha \rightarrow \alpha|t) = e^{-ut} + (1-e^{-ut})\pi_\alpha \tag{3.3}$$

3.2.3 Restrictions on Evolutionary Histories

Here, we describe two additional restrictions imposed by Indelign on the evolutionary histories considered. We first introduce some terminology regarding multiple alignments. (See Figure 3.3A for an illustration.) A sequence at a leaf node is called an extant sequence; sequences at internal nodes are called ancestral sequences. Any contiguous stretch of gaps in any extant sequence is called a "hole". Any column of the multiple alignment where a hole begins or ends in any sequence is called a "hole-boundary". Any stretch of alignment columns that is bordered by two successive hole-boundaries, or by the alignment boundary and an adjacent hole-boundary, is called a "block". Thus, a typical multiple alignment is a concatenation of multiple blocks. Two adjacent blocks are said to be "mutually-dependent" if both blocks contain a hole in the same sequence, e.g., in Figure 3.3A, blocks DE and EF are mutually-dependent, and so are blocks EF and FG.

The following two constraints are imposed on any valid IEH (Indel Evolutionary History) or CEH (Complete Evolutionary History) by Indelign. (IEH and CEH were defined at the beginning of Section 3.2.1.)

Assumption 1: *Any two aligned nucleotides in the IEH must share a common ancestral nucleotide.* In other words, neither of the two aligned nucleotides may be the result of an insertion event. (See Figure 3.3B, cases 1-3, for some examples of evolutionary histories prohibited by this assumption.) This assumption respects the implicit semantics of an aligned position being an evolutionarily orthologous position, and has been used in other related work [10].

Assumption 2: *An indel event begins and ends at hole boundaries.* This implies that if we align the ancestral sequences with the given alignment of the extant sequences, the hole boundaries and block locations will remain unchanged. (See Figure 3.3B, case 4, for an example. The single hole BD in the second species is not allowed to result from two separate deletions BC and CD, as per Assumption 2.)

Claim 1: If two adjacent blocks are not mutually-dependent, an IEH cannot have an indel event spanning the two blocks, *on any branch of the tree.* (This follows from the two assumptions above.)

3.2.4 Block Level Representation of Sequences

Let l and r be the boundary positions of a block. By Assumption 2 and Claim 1, the subsequence $S_v[l \ldots r]$, for any node v, is either a string of gaps, or a string of nucleotides (known for extant sequences, unknown for ancestral sequences). This allows us to rewrite all sequences (assigned to nodes in an IEH) at a block-level, as follows. Let $B = (l, r)$ denote the block from position l to r, $\{B_i\}_{i=1\ldots k}$ be the sequence of blocks in the multiple alignment, and $S_v[B_i] \equiv S_v[l_i \ldots r_i]$, which implies $S_v = S_v[B_1] \cdot S_v[B_2] \cdots S_v[B_k]$. We define

- $R_v[B_i] =$'$*$' if $S_v[B_i]$ is all '*'s, ('$*$' is an unknown nucleotide.)
- $R_v[B_i] =$'$-$' if $S_v[B_i]$ is all gaps, and
- $R_v[B_i] =$'N' if $S_v[B_i]$ is all nucleotides.

Finally, we rewrite the sequence assigned to node v as $R_v = R_v[B_1] \cdot R_v[B_2] \cdots R_v[B_k]$. Thus,

- an IEH is simply an assignment of each $R_v[B_i]$ as '$*$' or '$-$', $\forall v \in V_I$, $\forall B_i$, and
- a CEH φ is a pair $(\delta_\varphi, \mu_\varphi)$, where δ_φ is an IEH, and μ_φ is a mapping from each $R_v[B_i] =$ '$*$' to a string $S_v[B_i] \in \{A, C, G, T\}^{r_i - l_i + 1}$. (See Figure 3.4.)

3.2.5 Enforcing Assumption 1

To find the maximum likelihood IEH, we must only find the optimal labeling of each block at each internal node as a '$*$' or '$-$' and do not need to compute the substitution terms (A_e) of Equation 3.2. It implies that in finding the maximum likelihood IEH, we do not need to sum over all possible CEHs consistent with a candidate IEH. It is therefore a key result for the efficiency of our algorithm. The substitution terms of the likelihood (Equation 3.2) can be computed separately for each column of the alignment, using Felsenstein's post-order traversal algorithm, thereby completing the likelihood computation.

Assumption 1 also restricts the space of IEHs that we have to search in order to find the maximum likelihood solution. Two specific restrictions that follow from the assumption are: (i) For any block B, for any internal node v, if its two child subtrees each have a node labeled as non-gap, then v itself must be labeled with a non-gap. (ii) For any block B, if two nodes are

labeled as non-gaps, and one of them is an ancestor of the other, then there may be no gap-labeled node on the path between them. These two restrictions are enforced by fixing which of the internal nodes must be labeled as non-gap ($*$).

3.3 Algorithm

The Indelign program has the following components.

1. ANNOTATE: Given a multiple alignment, this obtains an IEH with maximum likelihood.
2. SEARCH: Given an initial multiple alignment, this makes limited changes to the alignment so as to obtain an IEH with maximum likelihood.

Overview

We describe below how the ANNOTATE component can find the maximum likelihood IEH either by naively evaluating all possible IEHs (Section 3.3.1), or by using an algorithmic technique called dynamic programming (Section 3.3.1). A crucial step here is to identify each block (or sequence of blocks) that can be annotated independently of other blocks, and to use a divide and conquer strategy that solves each subproblem (annotation of a sequence of blocks) separately before merging the solutions into an overall annotation. This is done without sacrificing correctness of the solution. The main idea behind the SEARCH component (Section 3.3.3) is to explore all alignments that may be obtained by making a limited class of modifications to the current alignment, find the best scoring among such "neighboring" alignments, and update the current alignment to this best scoring "neighbor". This update is then repeated as long as improvements are obtained.

3.3.1 ANNOTATE

The ANNOTATE algorithm can be run with or without complete knowledge of the model parameters. In Sections 3.3.1 and 3.3.1, we assume that the parameter values are known *a priori*. If not known, they are learned from the data, as described in Section 3.3.2.

Enumerative Algorithm

This simple algorithm enumerates all possible IEHs in the restricted search space by Assumption 1. It computes the likelihood of each IEH, and outputs the maximum likelihood IEH. For any block B, at any internal node v, there are at most two possibilities for the label $R_v[B]$ ('$-$' or '$*$'). In a full binary tree with n leaves, there are $n-1$ internal nodes. Hence there are at most 2^{n-1} possible IEHs for the block. Now consider a maximal sequence of blocks where every adjacent pair of blocks is mutually-dependent (e.g., blocks DE, EF, FG in Figure 3.3A). The maximum likelihood IEH of the entire multiple alignment must include the maximum likelihood IEH of this sequence of blocks, since no other blocks are mutually-dependent with them. (Claim 1 in Section 3.2.3 ensures that there can be no indel *on any branch* that straddles across the boundaries of this sequence of blocks.) There are at most $2^{k(n-1)}$ IEHs, where k is the number of blocks, and therefore the enumerative algorithm has complexity $\mathrm{O}(2^{k(n-1)}n)$. (The extra factor of n is due to the $\mathrm{O}(n)$ time complexity of evaluating each IEH.) The time complexity of the finding an IEH of the entire multiple alignment is the sum of the contributions from each such maximal sequence of mutually-dependent blocks.

Dynamic Programming Algorithm

Indelign also implements a dynamic programming algorithm to find the maximum likelihood annotation. As in the enumerative algorithm, it operates on each maximal sequence of blocks where every adjacent pair of blocks is mutually-dependent. (This follows from Claim 1 in Section 3.2.3.) Let k be the number of blocks in such a sequence. The algorithm must assign to each internal node v, a string $R_v \in \{*,-\}^k$. The dynamic programming algorithm associates with each node $v \in V_I$, a table DP_v with 2^k entries. Each entry of DP_v is indexed by a string $R_v \in \{*,-\}^k$, and records the likelihood of the maximum likelihood IEH for the subtree rooted at v *if node* v *was labeled with* R_v. Let e_1 and e_2 be the edges from v to its two child nodes c_1 and c_2 respectively. Given the labels $r_1 = R_{c_1}$ and $r_2 = R_{c_2}$ assigned for these k blocks to the two child nodes, it is straight-forward to compute the insertion and deletion events on each branch, and their likelihood contributions $I_{e_1}(R_v, r_1)D_{e_1}(R_v, r_1)$ and $I_{e_2}(R_v, r_2)D_{e_2}(R_v, r_2)$ as functions of the start label R_v and end label

(r_1 or r_2). The dynamic programming recurrence then is:

$$\text{DP}_v[R_v] = \texttt{max}_{r_1,r_2}[(\text{DP}_{c_1}[r_1]I_{e_1}(R_v, r_1)D_{e_1}(R_v, r_1)) \times (\text{DP}_{c_2}[r_2]I_{e_2}(R_v, r_2)D_{e_2}(R_v, r_2))]$$

Since there are at most 2^k possibilities for each of r_1 and r_2, the time complexity of this algorithm is $O(2^{3k}n)$. We note that enforcing Assumption 1 as described in Section 3.2.5 fixes the label at many internal nodes to be '$*$'. Hence, in practice, the number of valid labels at each internal node is much less than 2^k, resulting in a significant reduction of running time. (See Section 3.4.5.)

3.3.2 Estimation of Parameters

In the previous section, we saw how a maximum likelihood IEH is computed, given the model parameters. These parameters include: (i) the single parameter u in the F81 substitution model, (ii) the insertion and deletion probabilities p_I and p_D respectively (for each branch e), and (iii) the probability distributions on lengths of insertions and deletions. Here, we describe how the model parameters are estimated from an IEH. If the parameter values are not known *a priori*, ANNOTATE starts with an estimate based on the input alignment, then iteratively computes the maximum likelihood IEH and uses the computed IEH to re-estimate the parameter values, until convergence.

To estimate u, we take the two closest related sibling species from the phylogeny, and count the number of times ($N_{\alpha\beta}$) that residue α in the first species aligns with residue β in the second, for all α, β. Let t be the total branch length between the two species. The contribution of the substitution events to the log likelihood is $log \prod_\alpha \prod_\beta Pr(\alpha \rightarrow \beta|t)^{N_{\alpha\beta}}$. Using Equations 3.2 and 3.3, and differentiating with respect to u, we get

$$\sum_{\alpha,\beta\neq\alpha} (-N_{\alpha\beta})(e^{-ut}(1-\pi_\alpha)+\pi_\alpha) + \sum_\alpha N_{\alpha\alpha}(1-\pi_\alpha)(1-e^{-ut}) = 0$$

from which we can solve for e^{-ut} and hence for u.

Estimation of the indel length distributions is done by fitting the assumed form of the distributions on observed histograms of the lengths. For instance, for a Zipf (or power-law) distribution of lengths, the observed histogram is converted to a log-log scale and a linear regression is done to

obtain the parameters of the distribution. The current implementation of Indelign supports three indel length distributions: Zipf, Poisson, and Geometric.

To estimate the insertion and deletion probabilities p_I and p_D, we assume these probabilities for any branch e with length t_e to be proportional to t_e, i.e., $p_I = c_I t_e$ and $p_D = c_D t_e$, where c_I and c_D are constants of proportionality independent of the branch. (Other, non-linear relationships may also be implemented in the future.) We then count the numbers of insertions and deletions N_I and N_D from the parent of the two closest related sibling species to either species, and estimate the constants c_I and c_D using the equations:

$$c_I + c_D = \frac{N_I + N_D}{t_{e_{12}} L} \qquad \frac{c_I}{c_D} = \frac{N_I}{N_D}$$

where $t_{e_{12}}$ is the sum of the lengths of two branches connecting the two sibling species.

3.3.3 SEARCH

The SEARCH component of Indelign uses the ANNOTATE component to score a multiple alignment by its maximum likelihood IEH, and searches the space of multiple alignments for the highest scoring IEH. It begins with an initial alignment obtained from an existing multiple alignment program, fixes the highly conserved parts of the alignment to be immutable, and performs a hill climbing search by changing the locations of holes locally. Any existing multiple alignment programs such as Mlagan [18], MAFFT [76] or TBA [11] can be used to generate an initial alignment. Recall from Section 3.2.3 that a hole may straddle multiple blocks, e.g., in Figure 3.3A, the hole DF in the second sequence straddles blocks DE and EF. The part of a hole that belongs to one particular block is called a "block-hole". We next describe each step of the SEARCH algorithm in detail.

1. Start with an initial alignment of the input sequences.

2. Identify statistically significant "two-sequence anchors" in the alignment. These are ungapped blocks of alignment (between any two sequences) with statistically significant percent identity. Any pair of aligned nucleotides included in an anchor is forced to being aligned throughout the search. Entire columns that are "immutable" in this sense partition the multiple alignment

into independent "inter-anchor regions" that are processed independently.

3. Annotate the initial alignment based on a "weighted maximum parsimony" (minimum number of indel events scaled by inverse branch length, see Section 3.4.2) criterion to get an initial IEH, and obtain an initial estimate of model parameters from this IEH.

4. For each inter-anchor region, repeat until convergence:

   (a) Choose a leaf node to process, in a round-robin fashion.

   (b) In the sequence at the chosen leaf node, pick a block-hole at random. Consider alignments that may be obtained by sliding the block-hole to the left or right (without crossing the adjacent hole or the inter-anchor boundary). Such alignments form the neighborhood of the current alignment.

   (c) Compute the maximum likelihood IEH for each of these neighboring alignments, using the ANNOTATE component, and use the (maximum) likelihood as the score of an alignment.

   (d) Move to the highest scoring among the neighboring alignments. This is steepest ascent or greedy algorithm. (We also implemented a Metropolis-Hastings move, and found the performance to be significantly inferior to this greedy heuristic, for comparable running times.)

5. Compute IEH from current alignment, estimate model parameters, and loop to step 4.

3.4 Experiments on Synthetic Data

We first performed evaluation studies on synthetic data generated with the simulation program Dawg [21]. An ancestral sequence of length 4000bp was chosen at random, and evolved along the branches of an input phylogenetic tree T with n leaves. The parameters for the simulation included an indel to substitution ratio, set to 0.1, and an insertion to deletion ratio, set to 1:3. The substitution model was Felsenstein 81, and indel lengths followed a Zipf (power-law) distribution. The synthetic data experiments allowed us to evaluate and compare different methods under controlled settings.

3.4.1 Evaluation Measures

We used the following measures for evaluating the accuracy of any given indel annotation by comparing with the true annotation.

1. Annotation Agreement: This is the fraction of indels in the true alignment that are correctly positioned and annotated (as insertion or deletion). (Optimal score: 1.)

2. Indel Agreement: This captures how close the output numbers of insertions and deletions are to the true numbers. It is defined by the formula $\sqrt{\frac{(N_{It}-N_{Ie})^2+(N_{Dt}-N_{De})^2}{(N_{It})^2+(N_{Dt})^2}}$, where N_{It} and N_{Dt} are true numbers of insertions and deletions and N_{Ie} and N_{De} are the output numbers. (Optimal score: 0.)

3. Indel Ratio Ratio: This is the true insertion : deletion ratio, divided by the estimated insertion : deletion ratio, given by the formula $\frac{N_{It}/N_{Dt}}{N_{Ie}/N_{De}}$. (Optimal score: 1.)

These evaluation measures were computed over all species excluding any outgroup species that is diverged directly from the root species.

3.4.2 Baseline Annotation Method

We used a column-wise maximum parsimony heuristic as a baseline annotation method. Each column of the multiple alignment was independently annotated so as to minimize the total number of 1bp events (substitutions, insertions and deletions) on the phylogenetic tree, each event being weighted by the inverse of the length of the branch on which it happens. This maximum parsimony annotation was achieved by a post-order traversal algorithm, using an extended alphabet (A,C,G,T,$-$). Finally, we merged adjacent 1bp events of the same type (insertion or deletion) on the same branch to form longer indel events.

3.4.3 Evaluation of the ANNOTATE Algorithm

We first performed experiments to assess the performance of the ANNOTATE component, if the true alignment is given. Annotation was done with (mode "T") and without (mode "U") the knowledge of model parameters. The second mode uses automatic parameter estimation. These two modes of ANNOTATE were compared to the baseline method based on column-wise maximum

weighted parsimony. A three-node phylogenetic tree was used (roughly based on the phylogenetic relationship of *cis*-regulatory sequences of *D. melanogaster*, *D. yakuba*, *D. ananassae*), 20 data sets were simulated and the various evaluation measures (Section 3.4.1) were computed for the outputs of each annotation method. This entire procedure was repeated five times, with the phylogenetic tree being "shrunk" (the length of each branch was shortened) by a scaling factor $\sigma = \{0.5, 0.6 \ldots 0.9\}$. Scaling the tree had the effect of decreasing the divergence time among the species simulated.

Figure 3.5 shows the averages of the different evaluation measures, for ANNOTATE (modes "T" and "U") and the baseline, at different values of the scaling factor σ. The ANNOTATE algorithm shows a significant improvement in performance over the baseline method. While the performance in mode "T" (known parameters) is better than in mode "U" as expected, the difference is marginal, demonstrating that the algorithm is *correctly able to learn the parameter values from the data*. We also note that the actual values of the evaluation measures for Indelign (ANNOTATE) are very close to the optimal, *demonstrating the practicality of the algorithm* on data sets that were designed to mimic the evolutionary divergence of the sequenced *Drosophila* species.

We repeated similar experiments on a phylogeny with five species (data not shown). We again find a significant improvement in performance of ANNOTATE over the baseline method, close-to-optimal values of the actual evaluation measures , and an insignificant difference between the "U" and "T" modes. (The phylogenies used for these experiments were based on inferred divergence distances of sequenced *Drosophila* genomes.)

3.4.4 Evaluation of the SEARCH Algorithm

We deployed the experimental set-up of the previous section to assess the improvement made by the SEARCH component of Indelign over the output of a popular multiple alignment program, Mlagan [18]. The initial alignment was done with Mlagan, run with default parameters and a gap opening penalty of 400. This alignment was annotated using Indelign's ANNOTATE component, run in the "T" mode (parameters known). The SEARCH component was then invoked to refine the initial alignment, and the the optimal alignment reported was annotated using ANNOTATE.

Figure 3.6A,B,D,E shows the average values of the "Indel Agreement" score and the "Indel Ratio Ratio" score, for three and four species simulations, for a range of values of the scale factor

σ. We notice a substantial improvement in average "Indel Ratio Ratio" when using Indelign to realign Mlagan output, especially for four species simulations (Figure 3.6E). In this case, this evaluation measure, whose optimal value is 1.0, goes from an average of 1.38 (using Mlagan) to an average of 1.23 (using Indelign) at $\sigma = 1$.

We also observe a significant improvement in the "Indel Agreement" score, especially for four species data (Figure 3.6D). For instance, at $\sigma = 0.5$, the average of this evaluation measure, whose optimal value is 0, moves from 0.10 (using Mlagan) to 0.07 (after realignment). Thus, we see a clear improvement in annotation of insertions and deletions once the input multiple alignment has been realigned with Indelign. We also note that the per-position alignment accuracy is not changed in the results of the SEARCH component from its value in the initial Mlagan alignment (Figure 3.6C,F).

3.4.5 Efficiency of Algorithms

The ANNOTATE component implements two different algorithms - an enumerative strategy and a dynamic programming (DP) strategy. We used our experimental set-up to compare the running time of these two strategies, and as seen in Figure 3.7, the DP algorithm leads to a significant improvement in efficiency. In absolute terms, the ANNOTATE component using the DP algorithm takes (on a 3 GHz Intel Xeon workstation) about 45 seconds (and about 18 MB memory) on a data set of 100 Kbp sequence for each of 4 extant species.

3.5 Experiments on *Cis*-Regulatory Modules in *Drosophila*

Our earlier work [154] documented that a set of 76 *cis*-regulatory modules (CRMs) in *Drosophila* show an excess of insertions over deletions, while it is known from prior work [126] that neutral sequences in *Drosophila* have a clear excess of deletions over insertions (in the ratio of 8:1). The results of Sinha and Siggia [154] were based on alignments of *D. melanogaster* and *D. yakuba*, with *D. pseudoobscura* as outgroup, using Mlagan as the alignment program, and an in-house implementation of maximum parsimony heuristics for indel annotation. We first repeated the same experiments using Indelign for the annotation step, and confirmed that the insertion : deletion count ratio was indeed greater than 1 (data not shown).

Next, we tested the same phenomenon on a larger data set of 448 CRMs (total length: 474,054bp) obtained from the REDfly database, with *D. ananassae* serving as a less diverged out-group than *D. pseudoobscura*. The phylogeny used in Indelign is ((D.mel:0.08,D.yak:0.08):0.13,D.ana:0.22), and the insertion : deletion count ratio, over all CRMs, is plotted as a function of the gap opening penalty of Mlagan, in Figure 3.8. We find a ratio greater than 2 regardless of the gap penalty parameter of the alignment program. The same conclusion is reached even when ignoring all 1bp indels (data not shown). The predominance of insertions over deletions was also observed when considering the total lengths of indels rather than their numbers (data not shown). We interpret these results as a reliable confirmation of the claims made in Sinha and Siggia [154], that regulatory sequences in *Drosophila* are enriched in insertions over deletions, in stark contrast to the situation in neutral sequences. This translates into an evolutionary trend of expansion in the regulatory sequences (and perhaps in functional non-coding sequences in general), at least since the *D. melanogaster*-*D. yakuba* split. The earlier observation [154] that *D. yakuba* had a much higher insertion : deletion ratio than *D. melanogaster* was also confirmed in the more comprehensive assessment done here. (See Figure 3.8.)

3.6 Discussion

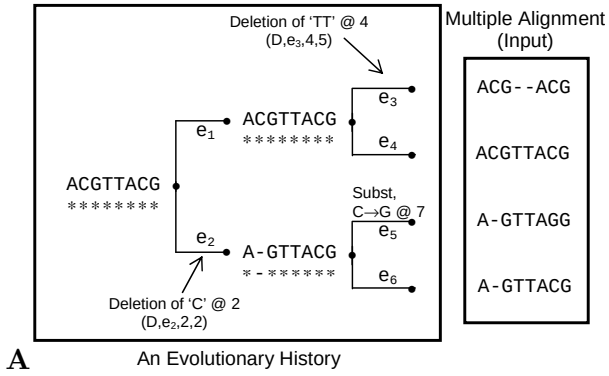
We have presented a probabilistic framework for the maximum likelihood annotation of insertions and deletions in an alignment of three or more species. To improve the annotation accuracy, the framework can also make limited changes to the alignment. The evaluations on synthetic data sets have shown that the framework is superior to the standard annotation method based on the maximum parsimony principle.

A limitation of the proposed probabilistic framework is that multiple indels at the same site are not allowed in the same branch. Such an extension would require major changes to the model and incur heavy computational overheads, and prohibit genome-wide applications. Moreover, we believe that such "multiple hits" of indels will be relatively uncommon when analyzing closely related genomes.

A direction for future research is to improve the SEARCH component of Indelign. While the current algorithm is able to provide improvements over the initial alignment, we noticed that it is

unable to make highly non-local changes to the input alignment.

3.7 Figures



A An Evolutionary History

B

	1	2	3	4	5	6	7	8	9	0	1	2	3
$S_{P(e)}$=	A	C	A	G	T	C	C	-	-	-	T	T	A
$S_{C(e)}$=	A	G	-	-	-	C	C	T	T	T	T	T	A

EVENT	REASON
(D,e,3,5)	$S_{P(e)}[3] \neq$ '-', $S_{P(e)}[5] \neq$ '-' $S_{C(e)}[3...5] =$ '- - -'
(I, e,8,10)	$S_{C(e)}[8] \neq$ '-', $S_{C(e)}[10] \neq$ '-' $S_{P(e)}[8...10] =$ '- - -'
(A,e,x) for x ∈ {1,2,6,7, 11,12, 13}	$S_{P(e)}[x] \neq$ '-', $S_{C(e)}[x] \neq$ '-'

Figure 3.1: (A) Complete evolutionary history (CEH) and indel evolutionary history (IEH) given a multiple alignment. Indel events are labeled as: I(nsertion)/D(eletion), branch, start, stop. (B) Examples of event specifications.

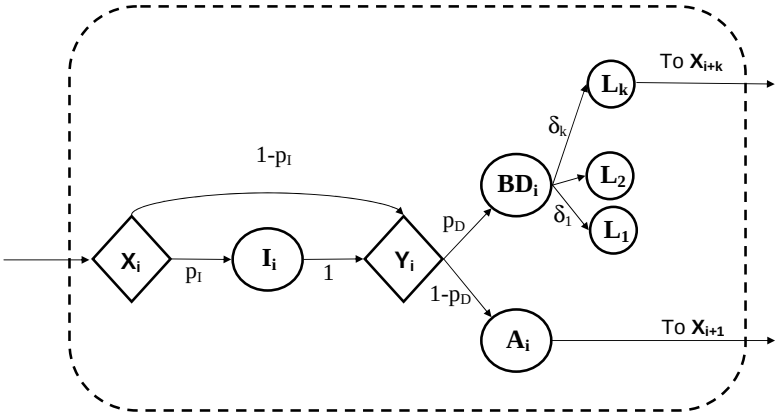


Figure 3.2: Hidden Markov model (HMM) structure of our evolutionary model.

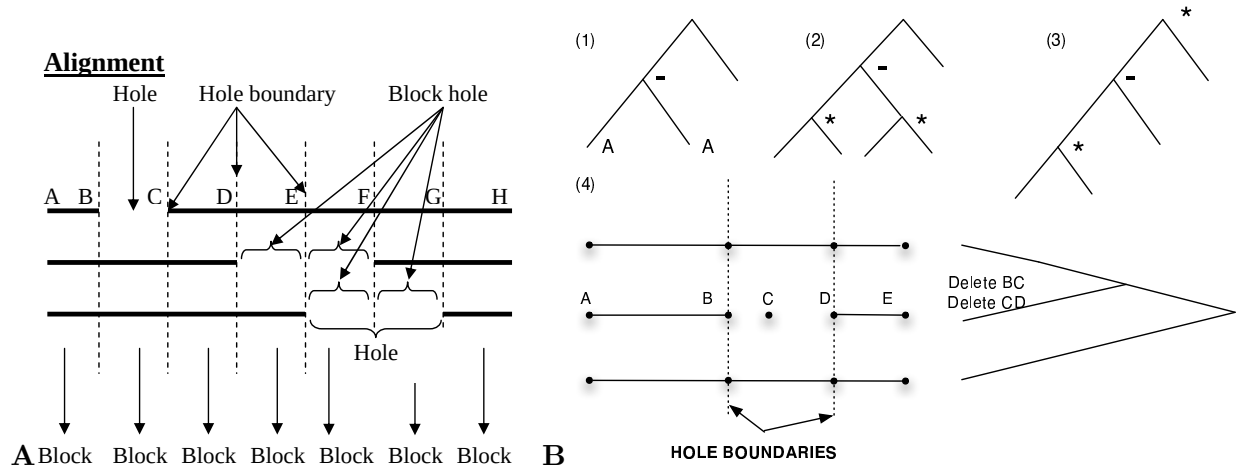


Figure 3.3: (A) Structure of a multiple alignment, (B) Evolutionary histories prohibited by Assumption 1 (cases 1-3) and Assumption 2 (case 4).

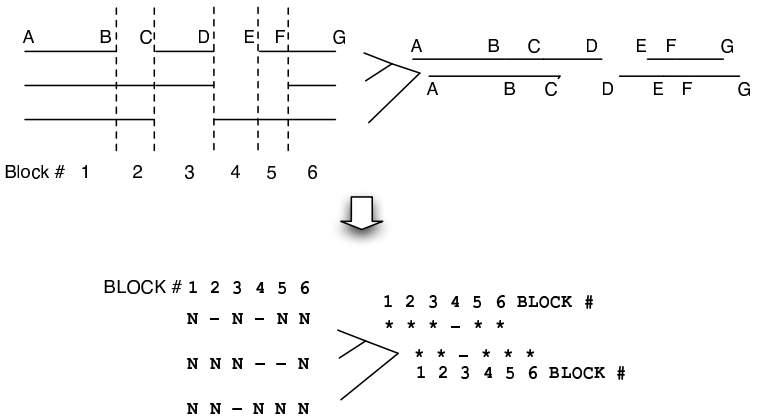


Figure 3.4: Block level representation of sequences. '$*$' is an unknown nucleotide, '$-$' is a gap.

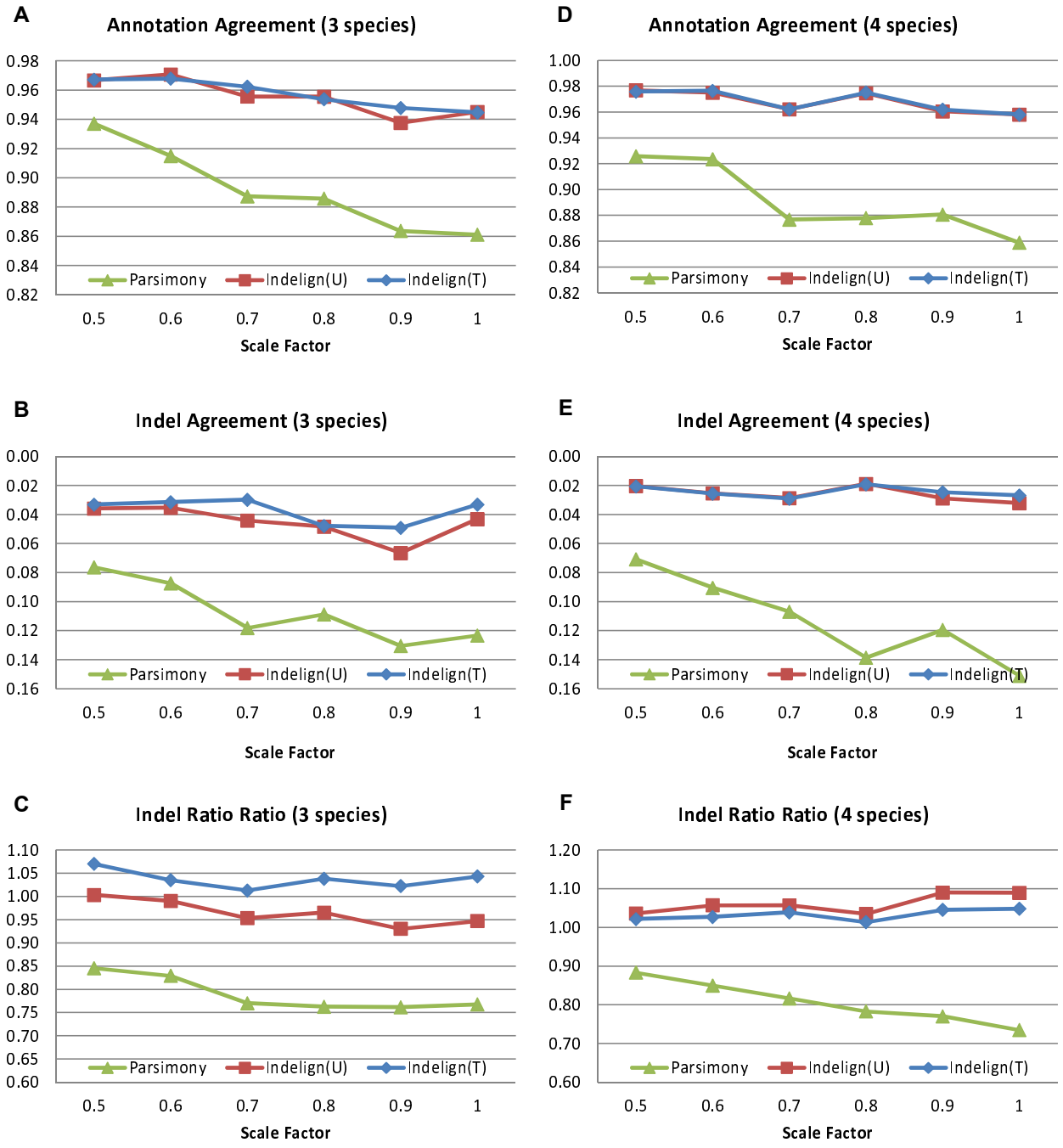


Figure 3.5: Performance of the ANNOTATE component of Indelign. Indelign was run with knowledge of parameters (T) or without (U) and a parsimony method on a phylogeny with three (A-C) and four (D-F) species. Each point is an average over 20 experiments. The default phylogenetic trees used in creation of data sets for three and four species are (spc1:0.08,spc2:0.08):0.13,spc3:0.22) and ((spc1:0.08,spc2:0.08):0.13,(spc3:0.08,spc4:0.08):0.13), respectively. In different sets of experiments, branch lengths are multiplied by different values of the "scale factor".

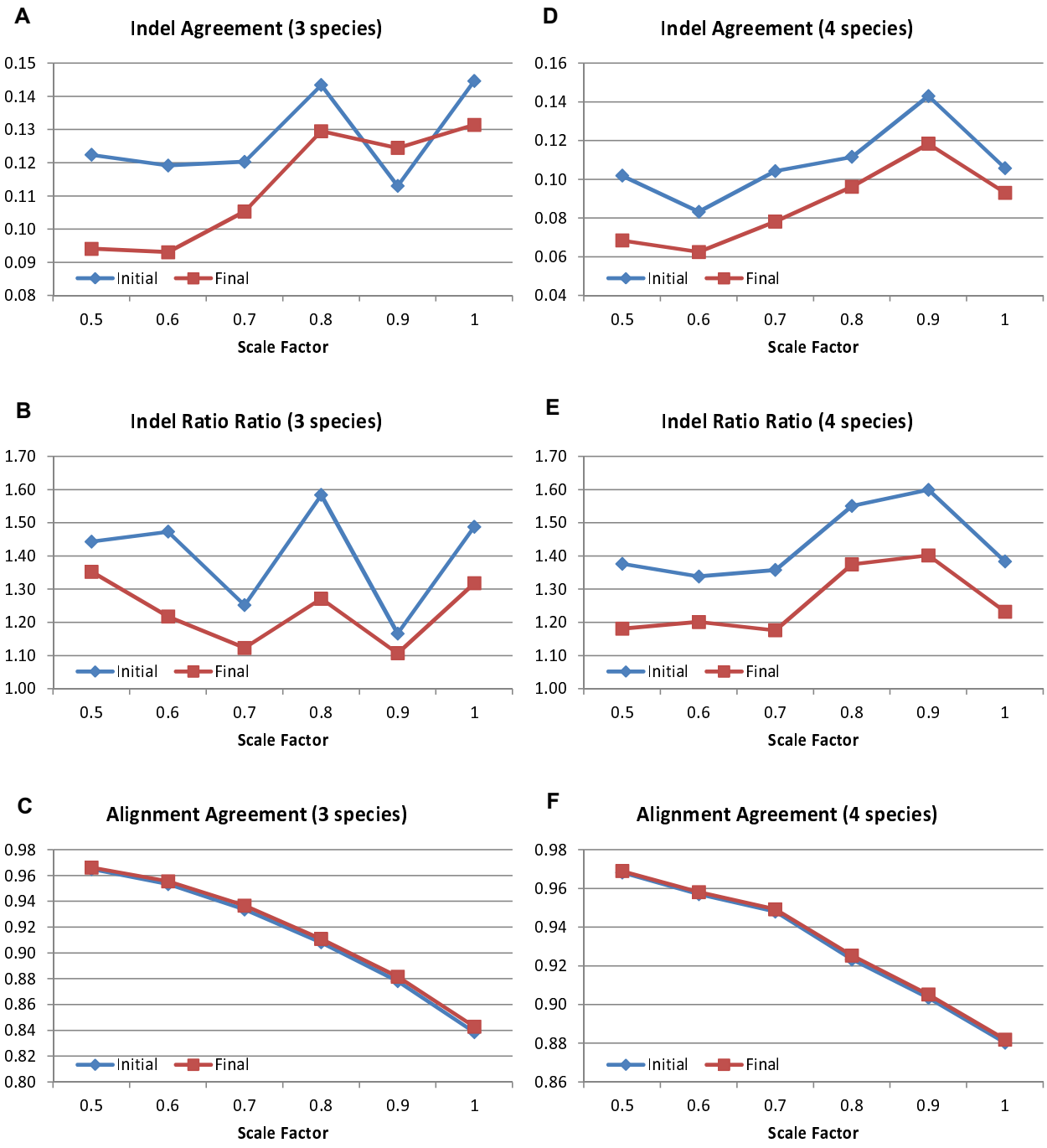


Figure 3.6: Performance of the SEARCH component of Indelign. The SEARCH component was evaluated by using the measures: 'Indel Agreement' and 'Indel Ratio Ratio' on phylogenies with three (A-B) and four (D-E) species. The Indelign annotation was compared between the initial (Mlagan) and final alignments produced by SEARCH. (C,F) 'Alignment Agreement' on a phylogeny with three and four species. The alignment agreement, defined as the fraction of aligned nucleotide pairs that are truly orthologous, is plotted separately for the initial (Mlagan) and final alignments. Each point is an average over 20 experiments. The phylogenetic trees used in creation of data sets of the three and four species are (spc1:0.08,spc2:0.08):0.13,spc3:0.22) and ((spc1:0.08,spc2:0.08):0.13,(spc3:0.08,spc4:0.08):0.13), respectively.

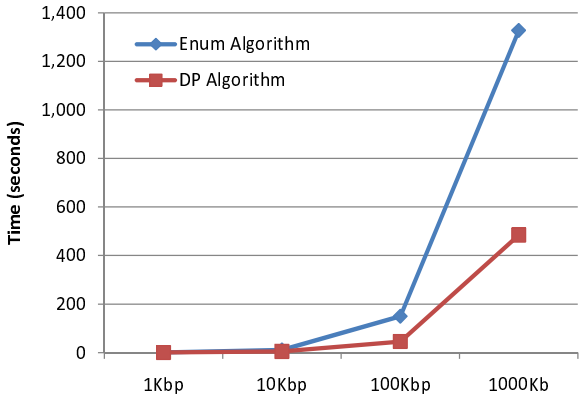


Figure 3.7: Running time of the enumerative and dynamic programming algorithms used by ANNOTATE.

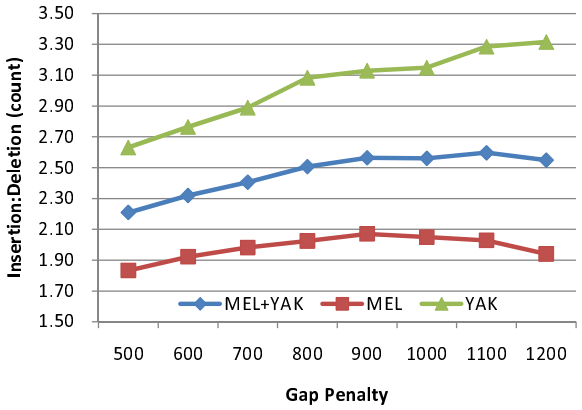


Figure 3.8: Ratio of raw numbers of insertions and deletions in the REDfly CRMs. The insertions and deletions are counted for *D. melanogaster* alone (MEL), for *D. yakuba* alone (YAK), and for both species together (MEL+YAK).

Chapter 4

Evolution of Regulatory Sequences in 12 *Drosophila* Species

4.1 Background

Gene regulation is well recognized as a major determinant of how an organism functions [30], and is also gaining recognition as an important evolutionary substrate [178, 181]. Transcription control is one of the most common forms of gene regulation, and is known to be implemented through regulatory sequences often in the neighborhood of genes. Binding of transcription factors (TFs) to certain positions within regulatory sequences enhances or inhibits transcription and these bound sequences are called transcription factor binding sites (TFBSs). In the case where a gene has to be combinatorially regulated by multiple transcription factors, the cognate TFBSs of those regulating factors tend to be clustered together in ∼1Kbp-length sequences called "*cis*-regulatory modules" (CRMs), or simply "modules" [67].

Despite significant recent efforts [3, 26, 98, 147], we lack a good understanding of the organizational principles of CRMs, e.g., the requirements on strengths and arrangements of binding sites within a particular CRM. Inter-species comparison of modules provides a major opportunity to improve our understanding of such principles: (i) Evolution of CRM sequences is constrained by functional requirements, so the study of CRM evolution should allow us to infer which underlying features are more important, and to what extent. (ii) One may hope to find certain evolutionary signatures of CRM sequences through careful inter-species analysis [160], greatly facilitating the identification of yet unknown CRMs. (iii) The study of CRM evolution will also enable us to better understand the path "from DNA to diversity" [20].

In this chapter, we study the relationships among orthologous *cis*-regulatory modules (CRMs) in 12 *Drosophila* species, especially with respect to the evolution of transcription factor binding sites, and report statistical evidence in favor of key evolutionary hypotheses.

Transcription factor binding sites are commonly predicted based on the assumption of their evolutionary conservation [93]. However, the exact nature of their conservation presents a complex picture. The study by Moses et al. [112] in yeast revealed that the rates of change of nucleotides of a TFBS depend on the binding profile of that TF - the positions of more specific protein-DNA binding permit lower rate of change. It should therefore be possible to leverage the position-specific substitution pattern to better predict TFBSs, as was done in [113]. This pattern has also been reported in bacteria [17] and vertebrates [104], but not in *Drosophila.* Given that this evolutionary pattern has already been assumed in practical analysis [180], it seems worthwhile to verify it in *Drosophila*. Moses et al. [113] further assumed that evolution of nucleotides at different positions are independent, and existing models of binding site evolution [56, 155] rely on this assumption; however, its validity is not obvious, given that a binding site typically functions as a unit. Empirical evidence either for or against this assumption has been lacking, except for a study in bacterial evolution [115] (where the evidence was against it). There is thus a clear need to test existing and new models of binding site evolution on the multi-species data from different phyla.

Even the most fundamental assumption of regulatory comparative genomics, that binding sites are evolutionarily conserved, has been challenged - Emberly et al. [43] found that binding sites are not substantially more conserved than their adjacent sequences in *Drosophila*; also, TFBSs are often found to have an unexpected amount of flux (gain or loss) in known CRM sequences [33, 38, 97] and in TF-bound regions in *in vivo* binding assays [12, 114]. It has been suggested that this flux is in part due to expression changes in the genes controlled by these sequences [12], and in part due to weak selection on individual sites even if the expression pattern of the target gene is conserved [96]. However, quantitative estimation of the strength of selection on binding sites has rarely been made, and requires extensive data on sets of orthologous binding sites. Moreover, the question of what leads to the observed levels of TFBS loss and gain is far from being resolved. For example, are the sites with higher binding affinities more likely to be conserved in evolution? How does the local context, i.e., the presence of other sites in the neighborhood, affect the probability of loss of a site? Does the loss probability correlate with overall selective pressure (substitution rate) of the CRM?

Cameron et al. [19] showed that insertions or deletions (indels) may be a powerful predictor of

CRM sequences in sea urchin, as long indels were suppressed inside CRMs relative to their neighboring sequences. Lunter et al. [99] speculated that such a selection pattern may be particularly relevant to CRMs, as the fitness of these sequences may be sensitive to the *length* of the sequences between adjacent TFBSs, but not their exact nucleotide composition. In several earlier studies involving a number of well-studied CRMs in *Drosophila*, such a pattern has not been fully observed [79, 97]. So the following question remains: is indel-purifying selection in regulatory sequences a general evolutionary force, common to different organisms? The answer will affect our understanding of CRM organization; e.g., how tolerant a CRM sequence is to the change of spacing between TFBSs.

Earlier attempts to characterize the evolutionary patterns of regulatory sequences used a few well-studied CRM sequences. These studies were limited in their scope [33, 79, 97]. The availability of 12 *Drosophila* species [40] and a large collection of experimentally verified *Drosophila* CRM sequences [53] enable a large-scale and more systematic study of the evolutionary patterns of CRM sequences. Such studies also crucially depend on accurate computational tools for sequence comparison. Commonly used multiple alignment tools [18, 110, 169] that treat regulatory sequences as no different from other types of DNA (or for that matter amino acid) sequences are known to be a source of errors in evolutionary analysis [143, 179]. Even if the alignments are accurate, the step of annotating gaps as insertions or deletions (usually done by *ad hoc* parsimony criteria) may lead to inaccurate inferences [95]. We have previously developed new methods for inter-species sequence analysis, that are specially designed with the properties of regulatory sequences in mind. These include (i) Morph [153], which optimizes pairwise sequence alignment by using the known binding profiles of relevant transcription factors, and (ii) Indelign [80], which uses a realistic probabilistic model of insertions and deletions to annotate "indel" events in a given multiple alignment. In this chapter, we take advantage of and extend these new methods to study the CRMs involved in *Drosophila* early development. This data set is ideally suited for such research because (i) the biological system is very well studied [8] and the relevant transcription factors are known, thereby limiting the false positives in binding site annotation, and (ii) much of the previous work on metazoan *cis*-regulatory evolution has been in this system [96, 97, 98]. Our study significantly extends the earlier work done on this dataset [90] and provides answers to many of the burning

questions alluded to above.

4.2 TFBS-Conscious Multiple Alignment and Binding Site Annotation

We begin with our findings on the evolutionary behavior of transcription factor binding sites. We collected 68 *D. melanogaster* CRMs and seven TF motifs, Bicoid (BCD), Caudal (CAD), *Drosophila* STAT (DSTAT), Hunchback (HB), Knirps (KNI), Krüppel (KR), and Tailless (TLL) from FlyReg [7] and the literature, involved in the control of anterior-posterior segmentation in the blastoderm stage embryo. These CRMs (source: REDfly [53]) have been experimentally determined, without using evolutionary conservation for discovery, and are hence suitable for evolutionary studies without introducing ascertainment bias. Orthologous sequences of these CRMs were extracted from 11 other *Drosophila* species and were aligned by a special multiple alignment program, called "ProbConsMorph". This is a new computational tool that we have developed, and is geared towards multiple alignments of regulatory modules in a TFBS-conscious manner. It avoids propagating pairwise alignment errors to the entire multiple alignment by combining the "consistency transformation" of ProbCons [37] with posterior alignment probabilities obtained from Morph [153]. We also repeated most of our tests using the alignment tool "Pecan" [122] that does not use TF motifs, and we point out differences, if any, between results from the two types of alignment.

We annotated binding sites for each transcription factor, in the subset of *D. melanogaster* CRMs that overlap with ChIP-bound regions from Li et al. [91], if such data was available. Site prediction was based on the p-value of match to the respective PWM ("position weight matrix") motif. We contrasted the density of these binding site predictions (in "bound" CRMs) with those in "unbound" intronic sequences, and typically found 2-3 fold enrichment in the former. We also predicted sites in each of the 11 other species separately, using the same method. Considering a binding site to be conserved if it is present in all other species in the *D. melanogaster* subgroup, we found that conserved sites were 2-3 fold enriched in CRMs than in intronic sequences. Our findings are consistent with earlier results in Li et al. [91], suggesting that the majority of predicted sites are likely to be functional.

Binding sites from different species, that overlap each other in the multiple alignment, are collectively referred to as an "orthologous TFBS set". Sites in such an orthologous set were re-aligned locally in order to correct for any errors in their precise alignment.

4.3 Binding Sites Have Position-Specific Substitution Rates

Different positions in binding sites have different contributions to the binding affinity of the TF. Positions that form the core regions for TF-DNA binding are more specific (less variation allowed) in the motif, and should be under stronger selective constraints. We thus expect different positions of TFBSs to have different degrees of evolutionary conservation. The specificity of a position can be expressed by the information content (IC) of the corresponding column in the PWM (position weight matrix), and the evolutionary rate by the number of substitutions in that position in orthologous binding sites. We observed highly significant negative correlations between specificity and evolutionary rate in five of seven TFs (i.e., all except CAD and TLL) (Table 4.1; Figure 4.1). Thus, our results confirm earlier similar findings in bacteria, yeast and vertebrates [17, 104, 112]. To avoid a bias introduced by the use of PWM-guided alignments, we used Pecan alignments of five closely related species for this particular analysis. The results were reproduced when using ProbConsMorph alignments (data not shown).

4.4 TFBS Turnover Follows a Molecular Clock

Even though TFBS loss and gain (henceforth called "turnover") have been commonly observed, it is not clear whether these changes are adaptive [2] or not [96]. If adaptive selection is the main force behind binding site turnover, it is likely that the process will show a lineage-specific pattern; on the other hand, a molecular clock has been known to be suggestive of the absence of adaptive selection, as per the neutral theory of evolution [83]. We considered the fraction of binding sites in *D. melanogaster* that have an ortholog (above threshold) in a second species, and plotted this fraction as a function of evolutionary divergence from the second species (Figure 4.2). For all transcription factors, the fraction of shared binding sites decreases linearly ($R^2 > 0.90$, Table 4.2) as the divergence time increases, a clear sign of a molecular clock. One problem that may

confound the analysis is the presence of false positive binding sites predictions, which are expected to follow a molecular clock. To examine this effect, we calculated a correction term in the fraction of conserved sites, and regressed this with divergence time, using the false positive proportion as a free parameter. Specifically, if F is the false positive proportion, then in a collection of n predicted sites, $n - nF$ are expected to be true sites, and if we observed m of the original n sites to be conserved, then an estimated $m - nFr$ of these sites are true sites, where r is the proportion of "false" sites that are conserved (estimated from intronic regions not bound in ChIP assays). This would give us a "corrected" conservation probability as $(m - nFr)/(n - nF)$. We performed regression analysis of this corrected conservation probability (versus time), while simultaneously estimating a false positive proportion F, i.e., leaving F as a free parameter. Adjusted R^2 values were computed as $1 - (n-1)(1-R^2)/(n-(k+1))$ where $k+1$ is the number of free parameters. (2 in our case-the false positive proportion and the slope of the line; the intercept was fixed at 1.) High values of the adjusted R^2 were obtained (Table 4.2), confirming the presence of the molecular clock. We repeated the exercise with sites for randomly created PWMs, and found a similar linear relationship. The rate of loss (negative slope of the line) for these random sites is higher than the rates for binding sites, for six of the seven transcription factors (Table 4.3), the difference being significant for BCD and KR. We note that the sites predicted by random PWMs do not represent neutral sequences, but reflect the average constraint in CRM sequences. This has been shown previously in [91]. The results were reproduced when using Pecan alignments (data not shown).

4.5 Evolution of TFBSs Is Affected by Binding Site Strength and Local Context

Having characterized some general patterns of TFBS evolution, in this section we study what specific factors may influence the conservation and turnover of binding sites.

4.5.1 Binding Site Strength

We defined the strength of a site as the degree of match of this site to the corresponding motif, as measured by a log-likelihood ratio (LLR) score [162]. TFBS turnover was defined as the number

of TFBS losses in unit evolutionary time (Figure 4.3). We focused on TFBS losses to avoid a possible ascertainment bias due to spurious binding site annotations when analyzing binding site gain events. We observed significant negative correlations between TFBS strength and turnover, for six TFs (i.e., all but CAD) (Table 4.4). We note that our alignment procedure, which uses motifs to construct and adjust the alignment, will tend to put strong sites in aligned positions and thus make them seem more conserved. We tested for such a bias by repeating the exercise with random PWMs that preserve the information content and G/C content of the original motifs. For each of the six motifs with significant correlations, their 100 randomized versions almost always had less significant correlations (see Table 4.4). The results were reproduced when using Pecan alignments (data not shown).

4.5.2 Constraint on CRM

Another potential determinant of turnover is the overall evolutionary constraint on the CRM to which the site belongs. We estimated the substitution rate for each CRM using the Paml software [183] and correlated it with the overall TFBS turnover rate of that CRM. We found a significant correlation only for one of the seven factors (DSTAT, p-value 0.0127), but this finding was not confirmed by the Pecan-based alignments.

4.5.3 Homotypic Clustering

A "homotypic TFBS cluster" [92] is a group of binding sites of the same TF, often found in the enhancers controlling early development in *Drosophila*. A homotypic TFBS cluster is thought to impart redundancy to the *cis*-regulatory apparatus, and should exhibit greater tolerance towards the loss of sites as compared to a CRM that has only one or two binding sites of the same factor. To test this, we computed the degree of homotypic clustering in a CRM as the number of putative *D. melanogaster* sites in the sequence (normalized by the sequence length), and correlated it to the overall TFBS turnover rate of the CRM. However, no significant correlation was observed (data not shown).

We next examined spatial proximity of homotypic binding sites at a finer granularity: if two adjacent sites of the same factor are closely located, there may be cooperative binding of the factor

to these sites, leading to stronger selective pressure. Such cooperative binding by proximal sites is known for the BCD transcription factor [89]. We calculated, for each TFBS, the distance to the closest site of the same factor and the distance was correlated with the turnover rate (Table 4.5). We found significant positive correlations for the factors CAD, HB and TLL. The only difference from Pecan-based alignments is the positive but weaker correlation for CAD (p-value 0.15), perhaps due to misalignments (see Section 4.7). Surprisingly, BCD did not exhibit any significant correlation in our tests with both types of alignments.

4.5.4 Proximity or Overlap of Heterotypic Sites

Binding sites often "interact" with sites of other factors in their neighborhood. Such interactions may include, for example, cooperative binding to DNA or short-range repression. We next examined the effect of spatial context of "heterotypic" binding sites on evolutionary constraint. In a procedure similar to that of Hare et al. [58], we classified sites as belonging to the "proximal", "distal" or "overlap" class depending on whether the closest site of another factor was within 10bp, more than 10bp away, or overlapping. We found sites in the "overlap" or "proximal" categories to be more conserved (present in all 12 species) as opposed to sites in the "distal" category (p-value 4.39×10^{-5} in ProbConsMorph based alignments, p-value 0.001 in Pecan alignments, Hypergeometric test).

We next tested the effect of the above spatial categories individually for each factor. Comparing the "proximal" and "distal" classes (Table 4.6, column "P vs D"), we found DSTAT and TLL sites to be significantly more conserved when having a proximal partner site (p-value 0.021 and 0.027 respectively). The same results were found using Pecan alignment (data not shown). Interestingly, BCD sites had a significant (p-value 0.012) tendency to be non-conserved (i.e., not present in all 12 species) if they had a proximal partner (Table 4.6).

In a similar comparison of the "overlap" class with its complement ("proximal" or "distal") (Table 4.6, column "O vs NO"), CAD, HB and KR sites showed a tendency to be more conserved when having an overlapping partner (p-values 0.002, 0.017, and 0.039 respectively). These results were reproduced when using Pecan alignments (data not shown).

In summary, five of the seven motifs showed a significant tendency to be conserved when they had a partner either overlapping with or proximal to them. KNI was the only motif examined

without such a property, and it is worth noting that KNI has the fewest sites in our collection. We also repeated the above test with the requirement that the "partner" site be that of a repressor (KNI, KR) when studying an activator TF (BCD, CAD, DSTAT) and vice versa. We found a significant result only for one TF (CAD, p-value < 0.05), and not for other factors, potentially due to small sample sizes (data not shown).

4.6 Deletions but Not Insertions Are Significantly Underrepresented in CRMs

Finally, we analyzed insertions and deletions in known regulatory sequences, to study the extent of indel-purifying selection. Among 370 non-overlapping *D. melanogaster* CRMs from the REDfly database [53], we chose 128 CRMs that have clear orthologous sequences in *D. simulans*, *D. yakuba*, and *D. erecta*. This choice of species was dictated by simulation-based assessment of the limits of our indel annotation capability. Because insertions and deletions (indels) may have different functional consequences on CRMs, we treat them differently. We estimated the number of short insertions and deletions in CRMs using Pecan [122] for alignment and Indelign [80] to annotate the indels. For each CRM, the insertion or deletion count was defined as the average of the respective counts in the four species, weighted by the branch length. We compared indel frequencies in CRMs to those in "background sequences", chosen to be the regions flanking the CRMs. We found that (i) the number of short deletions (less than 20bp in length) in CRMs is significantly smaller than that in background regions (paired Wilcoxon signed-rank test, p-value 0.0074; 1970 in CRMs and 2183 in length-matched background regions) and (ii) there was no statistically significant difference (p-value 0.5464) in the number of insertions (1932 in CRMs and 1870 in background). The number of long indel events (20bp or longer) in our data set was relatively small (CRM: 107 insertions and 175 deletions, background: 115 insertions and 178 deletions) and no significant difference was observed in this regard between CRMs and background regions.

Another related question is the indel pattern in the "spacer" region between CRMs and transcription start site (TSS) of the target genes. Transcriptional regulation depends on the communication between CRMs and promoter sequences [9], which may pose some requirements on the

length of the spacer sequences. We thus repeated the above analysis on these spacer regions. (We only consider 63 upstream CRMs in this experiment.) No significant differences in frequencies of insertions or deletions were observed between these regions and background sequences (data not shown).

Our results show that indel-purifying selection exists on CRM sequences, but such selection acts most strongly on deletions. We did not find clear suppression of long-indels, as has been observed before [19].

4.7 Discussion

The study of *cis*-regulatory evolutionary patterns has provided important insights on regulatory sequence function [16, 97], and proves valuable for prediction of these sequences in genomes [160, 182]. Yet, our understanding of *cis*-regulatory evolution is limited at best. While we have theories as well as a large volume of empirical data on protein evolution, we essentially have no theory and have made limited observations on the evolution of regulatory sequences. Our goal here is to begin to bridge the gulf between the vast amount of genomic sequence data and our poor understanding of regulatory sequences and their evolution. We have conducted a detailed evolutionary analysis of a large collection of experimentally verified CRM sequences, taking advantage of the recently sequenced 12 *Drosophila* genomes.

There are several technical issues that were important to address in our analysis. Evolutionary comparison depends on the alignment of orthologous sequences, but in general, alignments cannot be perfectly determined and may be a source of biased conclusion [179]. This may be a particularly serious problem for the analysis using 12 *Drosophila* species because of the relatively large divergence. We addressed this concern by developing a new multiple alignment program tailor-made for regulatory sequences. It combines the power of a pairwise regulatory sequence alignment tool, Morph [153], and a probabilistic multiple alignment framework ProbCons [37]. We have made this new software (ProbConsMorph) available freely for public use, to facilitate future studies of this genre. Nevertheless, the use of motifs to construct alignment may artificially boost the conservation level of TFBSs. We carefully addressed this potential bias whenever it may affect our conclusion. For example, when testing the positional variation of substitution rates, we use Pecan-based align-

ments without using motifs and limited ourselves to five closely related species. Similarly, when testing the correlation of binding site strength to turnover rates, we use randomized PWMs (as "negative controls") to validate our finding. We also repeated all our analyses with Pecan-based alignments. The various trends were almost always reproduced. One notable difference was that the correlation between nearest homotypic site distance and evolutionary rate (Table 4.5) for CAD was statistically significant (p-value 0.02) in ProbConsMorph alignments, but insignificant (p-value 0.15) in Pecan alignments. We suspect that this may be due to the tendency of standard alignment tools (such as Pecan) to misalign one or two nucleotides at the boundary of binding sites, especially if the motif contains short repeats such as TTTT [153], as is the case for CAD.

Another critical component of our analysis is the prediction of TFBSs. By using the same PWMs for all the genomes, we have made the assumption that the PWM of any TF is fully conserved across 12 *Drosophila* genomes. This is questionable, as researchers have found in yeast that the change of TF binding specificities can be an important part of the evolutionary change of regulatory networks [168]. For the seven motifs we analyzed, however, there is prior computational evidence that the binding specificities have not changed between *D. melanogaster* and *D. pseudoobscura* [156]. Another issue related to TFBS prediction is that predicted binding sites tend to have a high proportion of false positives [42]. We believe this problem is mitigated by our focus on the segmentation network, the fact that we restrict ourselves to transcription factors and CRMs experimentally known to be involved in regulating the segmentation genes, and our use of ChIP-based binding information wherever possible. We also believe that within a CRM, any computationally predicted binding site for a relevant transcription factor can "attract" transcription factor molecules, and contribute to the expression pattern, and should thus be considered "functional" in a broad sense. The results from Janssens et al. [71] seem to support this point. In practice, we may still have a small number of false predictions because of inaccuracies of the PWMs and we have attempted to estimate the false positive proportion by various methods. Also note that while false site predictions may obscure the evolutionary pattern of functional binding sites, they will not, in general, introduce spurious patterns (since, by definition, these sites are not under selection). In cases where the false sites may affect our interpretation of results, for example, in the test of molecular clock for binding site turnover, we have tried to make appropriate corrections.

In addition, in estimating TFBS turnover rates, we have emphasized on losses rather than gains of sites, because a predicted TFBS loss event has stronger supporting evidence than a gain event (the "gained" site is more likely to be a false positive prediction).

Our findings of a molecular clock extend earlier results on a small number of well characterized CRMs [28] across three *Drosophila* species, suggesting that this is a property common to developmental CRMs across a large evolutionary range. Even though we cannot exclude the presence of adaptive selection in individual cases, our results seem to suggest that negative selection to maintain the existing binding sites is the dominant mode of evolution, coupled with the occasional loss of sites due to random drift. The rate of site loss likely reflects the strength of purifying selection.

An unexpected result of our analyses is that the degree of homotypic clustering does not affect turnover rate. This is contrary to the notion that more binding sites of the same type will lead to greater redundancy, easing the selective pressure on the individual sites. Instead, the number of binding sites seems to be important to CRM function. In a more detailed analysis of homotypic clustering, now considering the binding site arrangement, we observed that for some factors, if a site is adjacent to another site of the same factor, this site will be less likely to be lost during evolution. This may be indicative of cooperative activity of proximal homotypic binding sites, leading to stronger selective pressure. For instance, the significant result (p-value 0.0184, Table 4.5) for CAD is consistent with anecdotal evidence of CAD sites being located as proximal pairs [31, 142, 146], although we are not aware of any biochemical evidence for such cooperativity. There is also some evidence in the literature for DNA binding by homodimers of TLL [36] and HB [120]. Our observation also suggests that sites that have a proximal "partner" are perhaps less likely to be spurious sites, which will provide a useful additional guideline to binding site prediction [29]. Surprisingly, we did not observe significant result for BCD, even though it is known to bind cooperatively [89]. This negative result is a reminder that the sensitivity of our statistical tests may be reduced due to a variety of factors, e.g., alignment errors, false sites, etc. These factors are unlikely, however, to produce spurious statistical signals.

We found that the presence of a binding site for a different factor, either overlapping or proximal to a binding site, can strongly affect the latter's evolution. Different mechanisms of local interactions between sites are known in developmental CRMs, e.g., cooperative binding between two

factors [157, 165], short-range quenching [52, 84], competitive binding to overlapping sites [157], etc. In all these cases, the loss of a single binding site may disrupt the interaction and create a larger change of expression than if the binding sites act in an additive fashion. As a consequence, these locally interacting site pairs may be under stronger selection. Our results support the importance of context in determining evolutionary fate of binding sites. A recent paper reports similar results for four CRMs of the *even-skipped* gene [58]. By working on a much larger set of CRMs, we confirm this context-dependence as a general evolutionary pattern. We also found some interesting specific cases, for example, the KR sites that overlap with another TF site, appear more conserved, consistent with the known role of KR as a repressor with the ability of competitive binding. In addition, the difference of the evolutionary patterns of the seven TFs suggests that they may depend on different mechanisms for their function. For example, both KR and TLL are repressors, but TLL is more conserved if it is adjacent to some other site, while KR is more conserved if it overlaps with another site. This seems to suggest that the relative importance of competitive binding and short-range quenching may be different in KR and TLL.

We did not find strong evidence of suppression of large indels within CRMs relative to their flanking sequences. Our results are different from an earlier study of indel patterns of CRMs in sea urchins, which reports that large indels (>20bp in length) are virtually absent inside CRM sequences [19]. There is an alternative explanation for this discrepancy: it has been known that *Drosophila* has a very compact genome as the neutral deletion rate is very high [128] and a large fraction (40-50 from different estimates) of intergenic non-coding sequences is under evolutionary constraint [2, 55]. Consequently, the flanking sequences of CRMs may not be entirely neutral, and the distinction between CRM and flanking sequences may not be as pronounced as in other species. (Our options were limited with respect to the "background" sequence to contrast with, since long repeats often used as neutral sequence in mammalian genomes [25] are rare in *Drosophila*.) The fact that short deletions are more constrained than short insertions is likely due to different effects of insertions and deletions on CRM sequences: any deletions that extend to an existing binding site will annul its functionality, while insertions, unless occurring exactly inside TFBSs, will only change the distance between sites, but not destroy them. These results combined with the lack of strong constraint on spacer sequences suggest that CRM structure is overall flexible, permits

relatively quick evolutionary change, and functions without being very sensitive to the precise distances between binding sites. In terms of its implications for bioinformatics, our results seem to indicate that the indel signature can be a useful CRM predictor but not strong enough to work alone, somewhat contrary to prior expectations [19, 99].

4.8 Figures and Tables

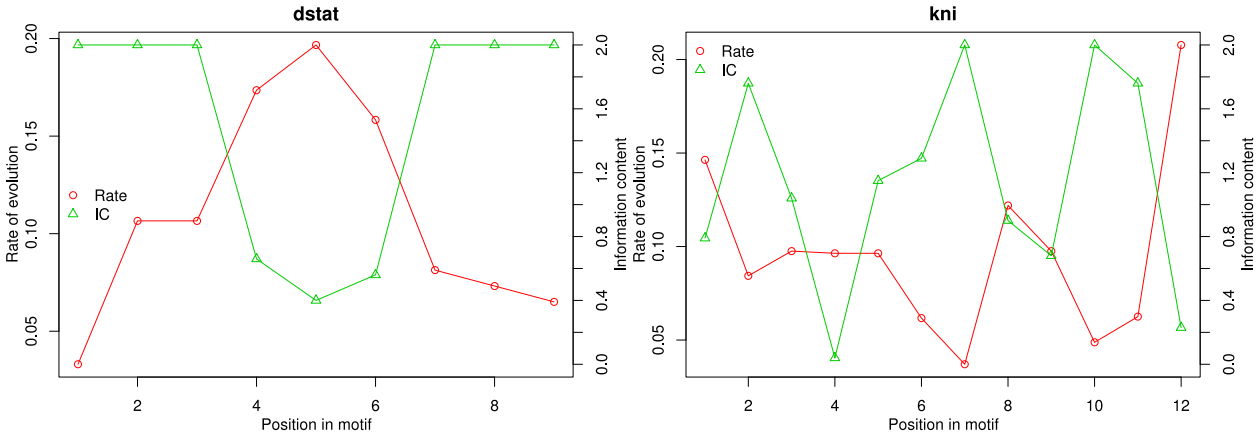


Figure 4.1: Correlation between the specificity of a TFBS position and its evolutionary rate in transcription factors DSTAT and KNI.

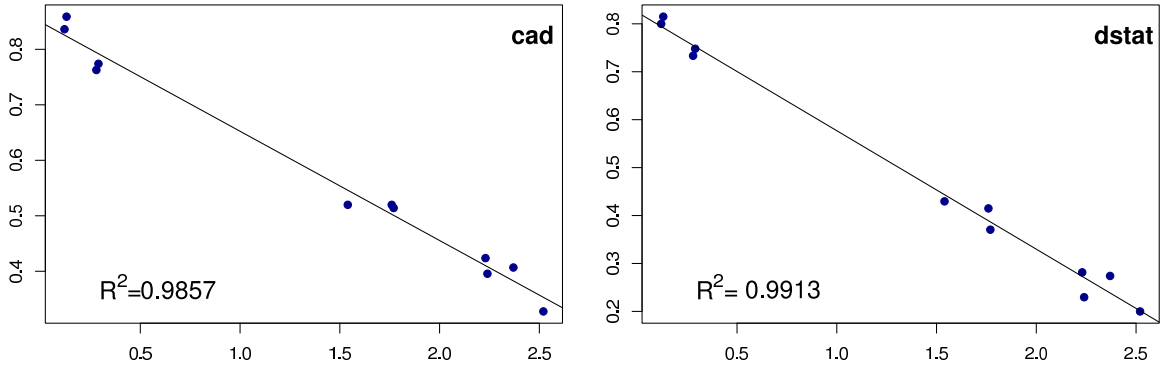


Figure 4.2: The fraction of *D. melanogaster* TFBSs that are conserved in a related species (y-axis), as a function of the divergence time to that species (x-axis), for transcription factors CAD and DSTAT.

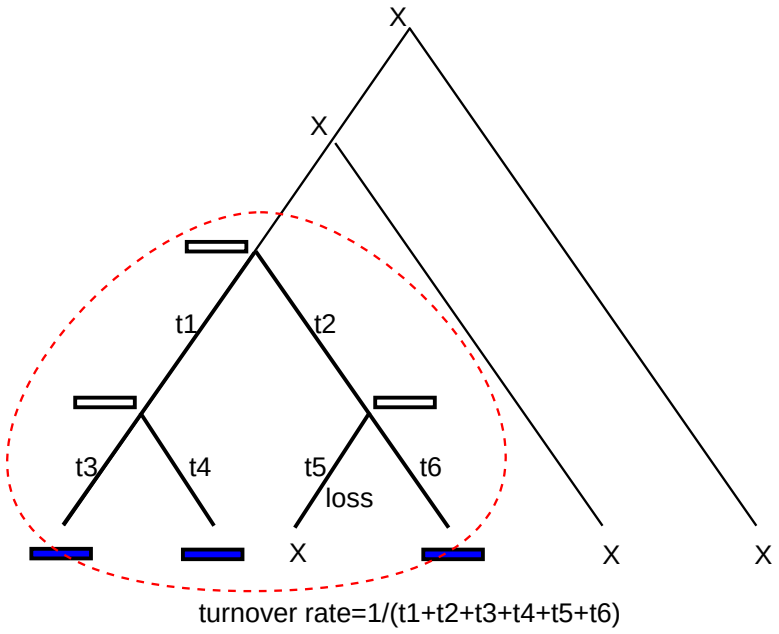


Figure 4.3: An example of the calculation of TFBS turnover rate. Among six species, only three species have a binding site (rectangles at the leaves). The subtree rooted at the least common ancestor of the binding sites is identified (in the dashed circle). There is one loss event in the subtree and, thus, the turnover rate is 1 (the number of events) divided by the sum of t1 though t6 (branch lengths in the subtree).

Table 4.1: Correlation between the specificity of a TFBS position and its evolutionary rate.

Factor	Number of TFBSs	Width of motif	Correlation coefficient[1]	P-value
BCD	160	8	-0.75	**0.0153**
CAD	175	9	-0.48	0.0969
DSTAT	129	9	-0.83	**0.0031**
HB	170	8	-0.69	**0.0347**
KNI	85	12	-0.82	**0.0005**
KR	177	11	-0.53	**0.0457**
TLL	185	10	-0.38	0.1375

[1]Spearman's correlation coefficient.

Table 4.2: Goodness-of-fit of a linear model for the fraction of conserved binding sites over divergence time.

Factor	R^2 (raw data)[1]	Adjusted R^2 (corrected data)[2]	FP[3]
BCD	0.9813	0.9631	0.14
CAD	0.9857	0.9693	0.29
DSTAT	0.9913	0.9831	0.26
HB	0.9114	0.9180	0.24
KNI	0.9642	0.9883	0.31
KR	0.9698	0.9097	0.27
TLL	0.9894	0.9515	0.32

[1] R^2 from raw data without correcting for the false positive rate.
[2] Adjusted R^2 from data corrected for the false positive rate.
[3] Estimated false positive rate obtained by regression.

Table 4.3: Comparison of loss rates of binding sites using real and random motifs.

		Random PWMs	
Factor	Loss rate	Mean	Stdev
BCD	0.1865	0.2530	0.0217
CAD	0.1969	0.2444	0.0213
DSTAT	0.2471	0.2642	0.0172
HB	0.1470	0.1937	0.0211
KNI	0.2315	0.2551	0.0170
KR	0.1811	0.2666	0.0172
TLL	0.2147	0.2389	0.0191

These rates are without false positive correction.

Table 4.4: Correlation between TFBS strength and TFBS turnover rate.

Factor	Number of TFBS sets	Correlation coefficient[1]	P-value	Random PWM[2]
BCD	163	-0.71	**0.0002**	0
CAD	168	-0.30	0.0974	18
DSTAT	129	-0.46	**0.0221**	11
HB	168	-0.62	**0.0030**	0
KNI	86	-0.58	**0.0025**	4
KR	191	-0.72	**0.0002**	0
TLL	188	-0.86	**<2.20E-16**	0

[1] Spearman's correlation coefficient.
[2] Number of random PWMs (out of 100 simulations) that show greater correlation than the real motif.

Table 4.5: Correlation between the distance between two adjacent homotypic sites and TFBS turnover rate.

Factor	Number of TFBSs	Correlation coefficient[1]	P-value
BCD	157	0.04	0.3969
CAD	162	0.38	**0.0184**
DSTAT	112	0.00	0.5000
HB	156	0.30	**0.0406**
KNI	82	0.24	0.1270
KR	183	0.14	0.2212
TLL	178	0.30	**0.0479**

[1]Spearman's correlation coefficient.

Table 4.6: Binding site conservation and its spatial context.

Factor	P vs D[1]	O vs NO[2]
BCD	0.9981*	0.5910
CAD	0.5626	**0.0015**
DSTAT	**0.0213**	0.8981
HB	0.2141	**0.0174**
KNI	0.4784	0.2425
KR	0.2071	**0.0387**
TLL	**0.0275**	0.0806

Numbers are P-values from hypergeometric test.
[1]P means proximal and D means distal.
[2]O means overlap and NO means non-overlap.
*The opposite p-value is 0.0124.

Chapter 5

Realistic Benchmarks for Multiple Alignments of Non-Coding Sequences

5.1 Background

With the continued development of new computational tools for multiple sequence alignment, it is necessary today to develop benchmarks that aid the selection of the most effective tools. Simulation-based benchmarks have been proposed to meet this necessity, especially for non-coding sequences. However, it is not clear if such benchmarks truly represent real sequence data from any given group of species, in terms of the difficulty of alignment tasks. Here, we develop a method to generate benchmarks for multiple alignments of *Drosophila* non-coding sequences, and show it to be more realistic than traditional benchmarks. Apart from helping to select the most effective tools, these benchmarks will help practitioners of comparative genomics deal with the effects of alignment errors, by providing accurate estimates of the extent of these errors.

The availability of genome sequences of closely related species (such as 18 placental mammal species [74] and 12 *Drosophila* species [24]) has provided opportunities to solve several key biological problems such as the inference of phylogenetic trees, reconstruction of ancestral genomes, estimation of evolutionary rates, identification of conserved and non-conserved regions, and more generally the study of genome structure and evolution. The alignment of multiple sequences, highlighting regions of homology among the sequences and predicting nucleotide level relationships among them, plays a critical role in such analyses. Numerous attempts have been made to develop accurate and efficient methods to solve the multiple sequence alignment problem (reviewed in [41, 118, 131, 152]), offering us much flexibility, as well as difficulty, in choosing the most appropriate tool(s) for the task. Another important task related to multiple alignment is the annotation of insertions and deletions (indels) in the alignment, a task that has received some attention in recent years [10, 13, 23, 35, 80, 158] in light of the realization that indels may be responsible for genomic variation as

much as nucleotide substitutions are [154], and that indels may affect regional mutation rates [175].

Given the availability of multiple tools to perform either of these two tasks, a researcher faces two important questions: "Which of the tools should I use for my task?" and "How accurate will the tool be on my data?" Answers to these come from studies that use data sets ("benchmarks") where the true answers are known, to evaluate and compare different tools. The design of benchmarks therefore directly affects the reliability of bioinformatics analyses that use those tools. The two most widely used benchmarking approaches for alignment tools are (i) to make use of biological sequences and their manually curated alignments from databases such as HOMSTRAD [109], BAliBASE [172], and SABmark [176], or (ii) to simulate the evolution of biological sequences by using specialized tools such as Dawg [21], Rose [163] and INDELible [49]. The main advantage of the former approach is the use of real biological sequences and alignments that are produced by using protein structure information. This approach does not apply to non-coding DNA sequences, whose alignments form the basis of regulatory comparative genomics. Therefore, simulation-based benchmarks have been widely adopted in this context [39, 100, 119, 132, 133, 144]. The simulation approach, however, is highly dependent on its parameters that reflect the underlying evolutionary processes and their rates. It is not clear how to choose "correct" settings for these parameters and how to assess if the simulated sequences mimic real data well enough for claims about alignment accuracy, both in relative terms (i.e., comparison of tools) and in the absolute, to generalize from the benchmarks to the real world setting. We address these questions in this work, whose main contributions are the following.

- We present a new simulation-based benchmarking method that is based on the entire spectrum of values of its parameters as inferred from real data. This is in contrast to existing approaches that rely on the average observed values of the parameters.

- We quantify the difficulty of aligning a data set by leveraging recent developments [86] on estimating alignment accuracy without requiring the "true" alignments. We reason that if the synthetic data sets truly mimic real orthologous sequences, the difficulty of aligning them ought to match that for the real data. This is the key insight used to determine how realistic a particular benchmark (i.e., collection of data sets) is, and we use this idea to show that the novel simulation method produces far more realistic benchmarks than the existing approach.

- Using our new benchmarks, we evaluate and compare the accuracy of six multiple alignment tools (ClustalW [88], Dialign-TX [166], Mafft [75], Mavid [15], Mlagan [18], and Pecan [124]) on *Drosophila* non-coding sequences. The specific alignment task we consider is that of global alignment of ∼1-10Kbp long sequences, and our conclusions may not apply to the task of local alignment, which was studied in [132]. We are able to estimate the accuracy of alignment for specific sets of *Drosophila* genomes, and find these to be very different from previously reported values. We also evaluate two schemes for annotating insertions and deletions specifically, and find their accuracy to be comparable, and close to optimal.

- We find that data sets with an excess of deletions over insertions are more amenable to accurate alignment than those with an excess of insertions, suggesting an implicit bias (in the alignment tools) with respect to their treatment of indels, even though none of the evaluated tools explicitly makes a distinction between insertions and deletions.

5.2 Simulation of Non-Coding Sequences by a Traditional Method

Modeling of DNA sequence evolution has been studied extensively in the past, and state-of-the-art simulation programs [21, 163] draw on various aspects of such models. Simulation of non-coding sequences [132] incorporates current understanding of the architecture of such sequences in terms of regions of evolutionary constraint, for example by stipulating the presence of short (but variable length) subsequences that evolve at a much slower rate than the rest of the sequence. We refer the reader to [21, 132] for a comprehensive description of these approaches, which form the foundation of our own work reported here. These simulation programs rely crucially on the values of their parameters (e.g., substitution rate or frequency of constrained blocks). The parameters serve to fully specify the stochastic processes from which evolutionary events (e.g., substitutions or indels) will be sampled, and prescribe the *expected* frequency of those events in the generated data sets. Variation in the frequency of these events, which underlie the difficulty of alignment tasks, results from the inherent randomness of the simulation process, i.e., the differences in random choices made from one "run" of the process to another. It is natural to ask if the resulting variability

across data sets in a synthetic benchmark is comparable to the corresponding variability observed in real orthologous sequences. The question is particularly relevant due to the heterogeneity of non-coding sequences with respect to the density of functional elements and also motivated by the known variation in evolutionary rates across loci [5, 99, 151].

We began by implementing the above-mentioned simulation paradigm, which we call the "traditional" paradigm, by incorporating the "constraint blocks" idea of Pollard et al. [132] into the Dawg simulation program [21]. Parameters, including phylogeny, branch lengths, indel frequency, and various parameters related to conserved blocks were set based on previously published values from the literature [6, 132] or estimated by us from published multiple alignments of *Drosophila* non-coding sequences, which can be downloaded from UCSC Genome Browser Database [74]. A key difference in our implementation was that branch lengths (i.e., average substitution rates) were estimated from non-coding sequences themselves, instead of synonymous substitution rates from coding sequences, as has been done previously. We elaborate on this important issue later in Section 5.5.2.

We considered the alignments of real *Drosophila* sequences from eight among total 12 *Drosophila* species: *D. melanogaster*, *D. simulans*, *D. yakuba*, *D. ananassae*, *D. pseudoobscura*, *D. willistoni*, *D. mojavensis*, and *D. grimshawi*, which were downloaded from UCSC Genome Browser Database [74] and consist of multiple segments of alignments in average 1Kbp length. We computed the sum of branch lengths of the phylogenetic tree estimated from the alignments, and found the distribution of this statistic to have a large variance across the genome (black bars in Figure 5.1). The same distribution, when computed from 100 synthetic data sets generated using the traditional simulator described above, and the same alignment program, shows a very sharp peak around the mean (dark gray bars in Figure 5.1). We note that the means of the two distributions are similar (1.87 in real data and 1.94 in synthetic data), since the benchmark was parameterized by the average substitution rates observed in real data. This is the first clear evidence that existing simulators fall short of representing the *range* of conservation levels in real data.

Since substitution rates are generally correlated with indel rates, a large variance in the former implies a corresponding variance in indel frequencies, which of course lie at the root of the alignment problem. This suggests that if we could measure the "difficulty of alignment" in any region of

the genome (e.g., by having knowledge of the true alignment, and measuring the accuracy of a powerful alignment program), we ought to see a large variability in this measure across the genome. Moreover, if the observed distribution of the alignment difficulty measure is comparable to that in a benchmark, we would be confident in making claims about performance of alignment tools based on that benchmark. The problem is that measuring alignment difficulty on real data requires knowledge of their true alignment, which is unavailable. Recent work by Landan and Graur [86] showed that a reasonable surrogate for the accuracy of an alignment program on a data set can be computed even without the true alignment. They reasoned that good alignments should be invariant to the *orientation* of the input sequences, and therefore defined the "Heads or Tails" (HoT) alignment quality score as the agreement between two alignments, one generated from original sequences and the other from their reversed versions. Hall [54] later showed that there is a clear positive correlation between HoT alignment quality scores and the real alignment accuracy measured by comparison with the true alignment. This remarkable finding inspired us to formulate the following strategy for quantifying the spectrum of alignment difficulty in data sets.

We computed the HoT alignment quality score on the computed alignment of a data set, and used this score as a surrogate for the alignment difficulty of the data set. (The alignment was computed using a well-established alignment program called Pecan [124], but other choices would not affect our conclusions.) Low values of the alignment quality score indicate that the data set is particularly hard to align, and high values are suggestive of an "easy" data set. As shown in Figure 5.2A, the distributions of the score were significantly different between synthetic and real data sets. Alignment quality scores for 83% of the synthetic sequences are above 95, whereas close to 50% of real sequences had scores below this range. This strongly suggests that by and large the synthetic sequences simulated by the traditional approach are easier to align than real sequences, even though the former were generated with evolutionary parameters mirroring their real data counterparts. In particular, the variance of alignment quality (and presumably of alignment difficulty) is much smaller in synthetic data sets.

5.3 Simulation Based on a Mixture Model of Parameters

We hypothesized that the above observation about synthetic data sets was due to the use of a single setting of the branch lengths, and the relatively low variability resulting from the randomness of the process itself (Figure 5.1). If this is true, then one way to alleviate the problem would be to allow for multiple phylogenies for simulation of different data sets, with the variability of branch lengths across phylogenies introducing an additional source of data set variability. We therefore considered a set of $K = 10$ phylogenies $\{\phi_1, \phi_2, \ldots, \phi_k\}$ that are scaled versions of the original phylogeny ϕ_0, i.e., every branch length in phylogeny ϕ_i is a constant factor τ_i times the corresponding branch length in ϕ_0. (We used $\{\tau_i\} = \{1, 2, \ldots, 10\}$.) We modified the simulator to first sample at random one of the K phylogenies, and simulate according to this setting of branch lengths, with all other parameters being fixed as before. In other words, the distribution of alignment quality scores from the new simulation process is a mixture distribution, with components parameterized by different phylogenies and the probability of sampling any particular phylogeny being the mixture weight. We estimated an upper bound on the agreement between this mixture distribution and the observed distribution of alignment quality scores, by maximum likelihood training of mixture weights, through Expectation Maximization [32]. This "best fit" mixture distribution is shown in Figure 5.2B, along with the real data distribution, and reveals a much stronger agreement between the two distributions, as compared to Figure 5.2A. The same trend was seen when allowing for a set of values of the "substitution to indel ratio" parameter (with values 10:1,10:2,...,10:5), keeping all other parameters, including the phylogeny, fixed (Figure 5.2C). These results strongly suggested that the use of a range of parameter values instead of a single value has great impact on the variability of alignment difficulty in synthetic data sets, and has the potential to lead to the generation of realistic sequences.

5.4 Simulation Based on Parameter Sampling

The above results, while encouraging in terms of better reproducing the genomic variability of alignment difficulty, were obtained by fitting parameters of the simulation process so as to best match real data. We next asked if we could achieve the same or better agreement between the

synthetic and real data distributions without having seen the real distribution of alignment quality scores. This would then allow us to use the observed agreement as a relatively unbiased assessment of how realistic the benchmark is. Developing the mixture model idea from the previous section, we now computed for each parameter the entire distribution of values observed in real data alignments, just as the traditional approach estimates the average of these values. The simulation process was now made to sample each parameter independently from its empirical distribution, and then generate a data set based on the sampled parameter values. The benchmark thus constructed (comprising 10000 different data sets) was examined for its distribution of alignment quality scores, and as seen in Figure 5.2D, this distribution was remarkably close to that observed in real sequences. In other words, the newly constructed benchmark meets our pre-specified criterion for a "realistic" benchmark. (It also shows strong agreement, as expected, with real data in terms of estimated branch lengths; Figure 5.1.)

The above analysis was performed using the sum-of-pairs score (SPS), which is the simplest of the scores defined in the HoT approach [86]. We repeated all analyses with another score, called the HoT column score (CS), and observed the same trends (Figure 5.3), although the agreement between synthetic and real data distributions was not as strong now as with the SPS (Figure 5.2D) (also see Section 5.7).

5.5 Assessment of Multiple Alignment Tools

5.5.1 Accuracy of Multiple Alignments

We used our new benchmark to evaluate and compare six leading multiple alignment tools that are publicly available and can align DNA sequences. These are ClustalW 2.0.5 [88], Dialign-TX 1.0.0 [166], Mafft 6.240 [75], Mavid 2.0 build 4 [15], Mlagan 2.0 [18], and Pecan 0.7 [124]. We performed the assessment with varying numbers of species, $K = \{3, \ldots, 8\}$. For each choice of K, 10000 sets of sequences corresponding to K different *Drosophila* species were simulated and the above alignment tools were run with default parameters or with the best setting recommended by their authors. We then compared the resulting alignments to the "true" alignments reported by the simulation program, using the following three commonly used evaluation measures [11, 14]:

(i) *alignment agreement,* which is the fraction of aligned base pairs (or bases aligned to gaps) in the predicted alignment that agree with the true alignment, (ii) *alignment sensitivity,* which is the fraction of aligned base pairs of the true alignment that agree with the predicted alignment, and (iii) *alignment specificity,* which is the fraction of aligned base pairs of the predicted alignment that agree with the true alignment. Whereas the alignment agreement score considers aligned base pairs as well as bases aligned to gaps, the sensitivity and specificity scores are calculated *only* from aligned base pairs. The results of our evaluations are shown in Figures 5.4, 5.5 and 5.6 (left panels).

The Pecan alignment program was found to be superior by all three measures, across all values of K. Its performance degrades more slowly (with increasing K) than the other tools, as a result of which the gap between Pecan and the other tools became larger more species were included in the tests. The average alignment agreement in five species alignments produced by Pecan (the species most divergent from *D. melanogaster* being *D. pseudoobscura*) was close to 80%, but degraded to $\sim$67% when aligning all eight species.

We performed the same evaluations by limiting ourselves to those data sets (in the benchmark) that had an excess of insertions over deletions, and separately to those data sets with an excess of deletions (Figures 5.4, 5.5 and 5.6; middle and right panels). Surprisingly, we saw a clear difference between these two classes of data sets, with most tools performing significantly worse when there was an excess of insertions in the data set. For example, on data sets with $K = 8$, ClustalW showed an alignment agreement of 36% or 46% depending on whether there was an excess of insertions or deletions (respectively). The same trend was seen in terms of the alignment sensitivity and specificity measures. Noticably, Pecan was largely unaffected by this dichotomy of data sets.

The evaluation measures used above consider all pairs of species in the K-species alignment and sum the accuracy values obtained from all pairs, without regard to the varying divergences of different pairs. In an attempt to address this issue, we separately measured the alignment accuracy of different pairs of species (e.g., *D. melanogaster* – *D. simulans*, *D. melanogaster* – *D. yakuba*, etc.), limiting ourselves to the eight-species data sets. All trends reported above were also seen in this alternative view of the results (Figures 5.7, 5.8 and 5.9). The alignment agreement, using Pecan, for *D. melanogaster* with *D. yakuba*, *D. anannassae*, *D. pseudoobscura* and *D. willistoni* was found to be 96%, 77%, 71% and 60% respectively.

5.5.2 Disagreement with Estimates Based on Existing Benchmark

We found a substantial disagreement between our performance estimates and those previously reported by Pollard et al. [132] using their own benchmark. For instance, the alignment sensitivity for the *D. melanogaster – D. pseudoobscura* pair comes out to be ∼70% in our assessment and ∼40% by their estimates, using the Mlagan alignment tool. We observe such gaps (with higher numbers in our benchmark) also for alignment specificity, and for other species pairs and alignment programs as well (data not shown). We confirmed this by evaluating the alignment programs ourselves on the Pollard et al. [132] benchmark. The benchmark generated by Pollard et al. [132] parameterizes each data set by a single value (substitutions per site) for the parameter, divergence distance. They provided estimate of this parameter value for the *D. melanogaster* and *D. pseudoobscura* pair (mean 2.4 and median 2.24) to link their simulations to the pair of species. They later updated this value in a new phylogeny (http://www.danielpollard.com/trees.html). We used their divergence estimates from the latter phylogeny and the benchmark they prescribed for this level of divergence, and evaluated the alignment programs ourselves on this benchmark.

While this discordance could be in part due to the fact that our benchmark employs a spectrum of parameter values to achieve greater and more realistic variability, we believe the major difference here is that even the average substitution rate, a key parameter in both simulation programs, is widely different between their study and ours. The estimate used by Pollard et al. [132] (∼2.4 substitutions per site) is based on silent positions in codons, while our estimate (∼0.38 substitutions per site) reflects the average substitution frequency (between these species) seen in non-coding sequences. In light of the results of Figure 5.2D, where we show that our benchmark accurately mirrors the range of alignment difficulty in real data, the use of non-coding sequences in estimating this key parameter seems better justified. We investigated this issue with additional tests. We collected data sets representing the *D. melanogaster – D. pseudoobscura* pair from Pollard et al. [132], as well as from our benchmark and the real genomes. The alignment quality score (HoT SPS) distributions were computed for each type of benchmark, and are shown in Figure 5.10. We observed a close agreement between our data sets and the real orthologous sequences, while the Pollard et al. [132] data sets were harder to align on average, consistent with the greater substitution rate used there. The overall substitution frequency observed in non-coding sequences

may be viewed as an average of the corresponding frequency in conserved blocks and the much higher frequency outside conserved blocks. This average is determined by two key parameters α, the fraction of sequence length that falls into conserved blocks, and β, the ratio of the evolutionary rate of conserved blocks to that outside blocks. Let t_o to be the overall evolutionary rate for both the conserved and outside blocks (the estimated branch length in a phylogenetic tree), t_n be the unconstrained evolutionary rate for the outside blocks (values provided to the simulation program). Then we have:

$$t_o = \alpha \times \beta \times t_n + (1 - \alpha) \times t_n$$

Given that the divergence estimate used by Pollard et al. [132] for these two species is ∼2.24 (median) substitutions per site, if we are to treat this value as the neutral rate (i.e., rate outside conserved blocks) in non-coding sequences, what values of α and β would lead to the observed overall substitution frequency of 0.38? We determined that if $\beta = 0.1$, as was used by Pollard et al. [132] (and also by us), α has to be ∼0.92, i.e., about 92% of non-coding sequences have to be conserved blocks, which is far higher than most current estimates of this parameter [6, 105]. Similarly, if we are to trust the values of $\alpha = 0.2$ and $\beta = 0.1$, as was used by Pollard et al. [132] (and also by us, based on estimates from real data), then the overall divergence, after averaging between conserved blocks and non-blocks, would be ∼1.84 substitutions per site, far greater than what is observed. We therefore concluded that the use of synonymous substitution rates as the neutral rate for non-coding sequence is likely to lead to benchmarks with overly "diverged" sequences that are more difficult to align than real sequences from those species.

5.6 Assessment of Indel Annotation Schemes

Traditional alignment programs mark the predicted locations of insertions and deletions as "gaps", and do not proceed to annotate these gaps as being insertions or deletions. This latter task has received some attention recently with at least two "indel annotation schemes" being published, based on maximum-parsimony ("sbInfer" [10]) and probabilistic-models ("Indelign" [80]), respectively. We examined the accuracy of these two alignment-related tools on our new benchmark. Here, we used a modified Indelign for additional efficiency. Specifically, the time complexity of

the Indelign program is exponential in the number of "conditionally dependent blocks" and this prohibits fast annotation of certain data sets with relatively large numbers of species [13]. To reduce the time complexity, when there are more conditionally dependent blocks than a predefined threshold, the alignment is heuristically partitioned by a block that has the smallest effect on the final indel annotation. This process is repeated until all dependent blocks with size greater than the threshold are resolved.

We noted that the best alignment agreement score (among all methods, as shown in Figure 5.7) is ∼70% for *D. melanogaster – D. pseudoobscura*, and decreases to ∼60% when a more diverged species (*D. willistoni*) is added. Reasoning that phylogenies for which computed alignments are largely inaccurate would not be suitable for insertion/deletion annotation in any case, we chose to limit our assessment to the following five *Drosophila* species: *D. melanogaster*, *D. simulans*, *D. yakuba*, *D. ananassae*, and *D. pseudoobscura*. The "true" alignment (as indicated by the simulation program) was provided to the two indel annotation tools and the insertion/deletion annotations on each of the five terminal branches (leading to the extant species) of the phylogeny were compared to the "true" annotations. The following three measures were used for assessment, borrowed from [80]: (i) *Indel Count Agreement*, which is the agreement of indel counts between true and predicted annotations, (ii) *Indel Ratio Agreement*, which is the agreement of the ratio of the number of insertions to the total number of indels between the two annotations, and (iii) *Indel Annotation Coverage*, which is the fraction of indel positions on which the two annotations agree. (Both sensitivity and specificity scores were calculated for the Indel Annotation Coverage.)

As summarized in Table 5.1, Indel Count Agreement scores of the two tools were very similar to each other and close to optimal (0) for most species except *D. pseudoobscura*, the species with the longest terminal branch in the phylogeny. Indel Ratio Agreement scores of both tools were close to optimal in all five species. While the sensitivity scores of Indel Annotation Coverage of the two tools were above 90% across all five species, the specificity scores were above 90% only for the four species except *D. pseudoobscura*. The loss of accuracy on the *D. pseudoobscura* branch is presumably due to the fact that there is no "outgroup" species to aid disambiguation of insertions and deletions on this branch. We further discuss the implications of these observations in Discussion. We also repeated our assessment for sequences with an excess of insertions or of deletions, as above, but no

significant differences was observed between these two categories (data not shown).

5.7 Discussion

Choosing the most suitable tool for aligning orthologous sequences is essential to studies in comparative genomics and in molecular evolution, making it critical to develop accurate benchmarking methodology. In this study, we propose a novel simulation-based approach to generate realistic data sets mimicking orthologous non-coding sequences from multiple *Drosophila* species. This new simulation method exploits the spectrum of values of evolutionary statistics (e.g., substitution rate, indel frequency) seen across a genome.

We take advantage of an objective "alignment quality" measure to show that the synthetic sequences produced agree with real sequences not only in terms of evolutionary statistics, but are also as easy or hard to align as real data sets. In this sense, our evaluation results are more likely to reflect the actual accuracy values of alignment-related tools on data from *Drosophila* species. We note that our strategy of sampling parameters (used in evolutionary simulations) from their empirical distribution has parallels with traditional Bayesian inference where one integrates over (i.e., samples from) a prior distribution on parameters, rather than using a single point estimate.

A key step in our benchmark construction was the ability to assess the quality of an alignment without access to the corresponding true alignment. This ability has been the result of several recent publications by other authors. Prakash and Tompa [134, 135] developed statistical methods to assess if a multiple sequence alignment appears contaminated with one or more unrelated sequences, based on which they identified regions of whole genome alignments as being suspect. The development of the "HoT" method by Landan and Graur [86] then came as a breakthrough to assess the reliability of multiple sequence alignments.

While our benchmark is shown to be very close to real sequences in terms of the distribution of HoT SPS, we are cautioned by the discrepancy observed between simulated and real sequences in terms of the HoT CS, an alternative alignment quality score from the same authors (Figure 5.4). This is likely the product of properties of non-coding sequences that are not adequately represented in our simulation process. For example, modeling the functional constraints embedded in non-coding sequences through short conserved blocks (with scaled down phylogenies) is surely an

oversimplification of the complexity of genomic architecture. Important progress has been made on this front, in the form of specialized evolutionary simulators that model transcription factor binding site evolution in realistic ways [60, 68, 133]. Each of these simulators makes specific assumptions about *cis*-regulatory architecture, vis-a-vis the density and evolution of binding sites. However, it is not yet clear which, if any, of these different assumed models of regulatory sequence evolution is most suited to represent the variability in constraint patterns across different regions of the genome. Our simplistic "conserved block" model (borrowed from [6]) seems to be a good approximation that captures the most prominent patterns in orthologous non-coding sequences, in terms of alignment difficulty. We expect that future research on more realistic models of *cis*-regulatory architecture will lead us to replacing the alternating arrangement of conserved blocks and faster evolving segments with a pattern more in line with reality. Future work may also include careful modeling of genomic repeats and repeat generating evolutionary events, since repeat-rich genomes may present additional challenges for the alignment task. Our proposed framework of sampling evolutionary parameters before running the simulation process will remain equally important in future benchmarks that implement such sophisticated models.

Some clarification is in order with respect to our manner of choosing substitution rates for the simulation process, since it marks a significant departure from traditional thinking. The latter, as embodied in the work of Pollard et al. [132], prescribes that the "unconstrained" parts of the sequence evolve with nucleotide substitution rate equal to that inferred from synonymous mutations in the nearby gene (or average over all genes). This rate ($\sim$2.4 substitutions/site for *D. melanogaster – D. pseudoobscura*) is widely different from the value observed in real non-coding sequence alignments ($\sim$0.4 substitutions/site). One could argue that this gap may be offset if we set an appropriate frequency of conserved positions (with very low rates), resulting in an average substitution rate that is close to the empirically observed value. However, this turned out not be the case for any realistic setting of the frequency of conserved positions (data not shown). We therefore chose to be guided by existing estimates of the frequency and length distribution of conserved blocks, with substitution rates that are some constant β times the "neutral" rate outside of the blocks, and set this neutral rate so that the average rate for the entire sequence matches observed values. Our choice reflects the philosophy that simulated data sets ought to match real

data in terms of various evolutionary statistics and net alignment difficulty, and the discordance of the used neutral rate from synonymous substitution rates is ignored for the sake of practicality.

To our knowledge, no previous benchmarking study has evaluated the effect of insertions and deletions on the performance of alignment tools. Some studies [100, 119, 132, 133, 144] have used equal frequencies for insertions and deletions and focused on the collective effects of indels. Here, we attempted to elucidate the differing effects of insertions and deletions by separately summarizing results for the two extreme cases where the number of insertions is at least two times the frequency of deletions and vice versa. The results were surprising, and indicated that most multiple alignment tools find it harder to accurately align data sets with an excess of insertions than those with more deletions (Figures 5.4 and 5.7). Löytynoja and Goldman [94] offered valuable insight into a possible source of this asymmetry, pointing out that progressive alignment methods (a category to which all the methods tested here belong) "end up penalizing single insertion events multiple times". We speculate therefore, as they did, that claims about insertion/deletion frequencies along the genome should be preceded by an examination of the alignment method's accuracy in regimes of high insertion frequency.

Finally, a note about our findings on insertion/deletion annotation. Indelign [80] is a probabilistic tool that annotates insertions and deletions by maximum likelihood training of an evolutionary model. sbInfer [10] is a greedy algorithm that reconstructs ancestral sequences based on the maximum parsimony principle, and therefore allows us to infer insertion/deletion annotations. To assess these two tools without being confounded by errors of an alignment program, we examined their performance on the true alignments. We found the two programs to have comparable accuracy on our benchmark for the five *Drosophila* species. While the accuracy was close to optimal on four of the five terminal branches, we observed that both tools over-estimate insertions as well as deletions on the longest branch (leading *to D. pseudoobscura*), while accurately predicting the ratio of insertions to deletions. We note that the *D. pseudoobscura* branch in the phylogenetic tree originates from the root of the tree, and we would expect to have better annotation results for this branch if an appropriate outgroup species was used. For studies that intend to use insertion to deletion ratio profiling to identify loci with unusual evolutionary patterns (e.g., [158]) it may be safe to examine all five terminal branches of this tree; however, for the more common requirement of accurately

annotating insertion and deletion events, e.g., to study gain and loss patterns of specific classes of transcription factor binding sites [81], we do not recommend using events on the *D. pseudoobscura* branch.

5.8 Figures and Tables

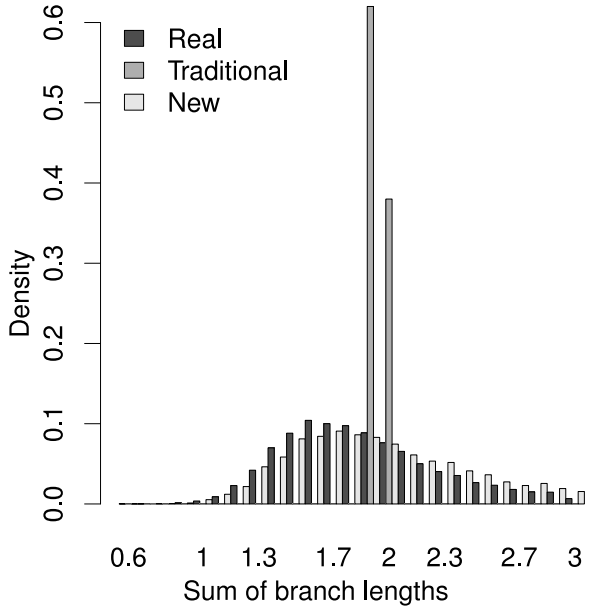


Figure 5.1: Distributions of sum of branch lengths in a phylogenetic tree estimated from real data and synthetic data respectively. Sequences of eight *Drosophila* species were collected from real data ("Real"), data produced by a traditional simulator ("Traditional"), and data produced by the new simulator based on parameter sampling ("New"). The traditional simulator used the average substitution rates observed in the real data, while the new simulator used the empirical distribution of substitution rates in real data. The branch lengths were estimated by Paml [183].

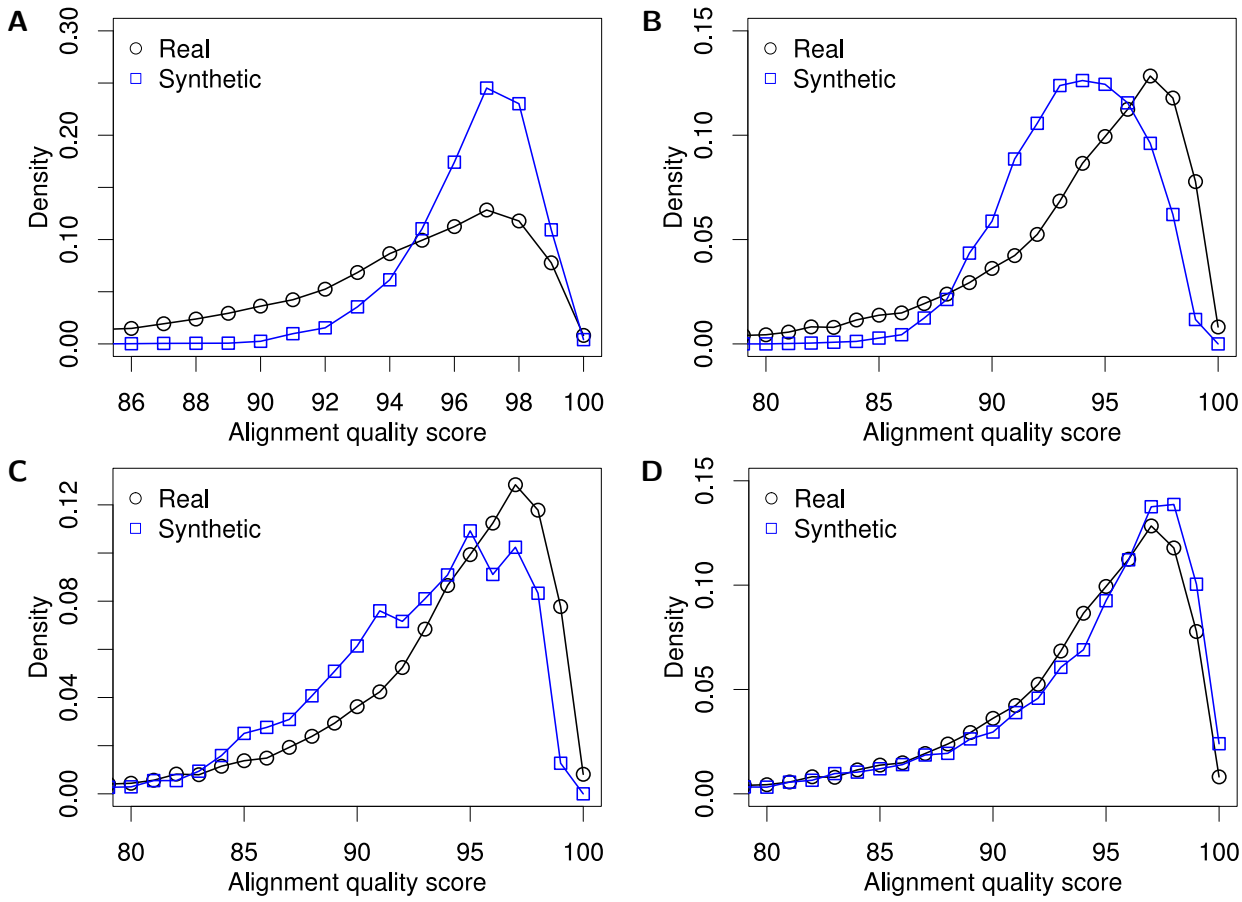


Figure 5.2: Distributions of alignment quality scores (HoT SPS) between real and simulated sequences. Synthetic sequences were simulated by (A) a traditional method, (B) using a mixture model of evolutionary rates, (C) using a mixture model of ratios of substitutions to indels, and (D) a novel method that relies on observed genome-wide distributions of its parameters.

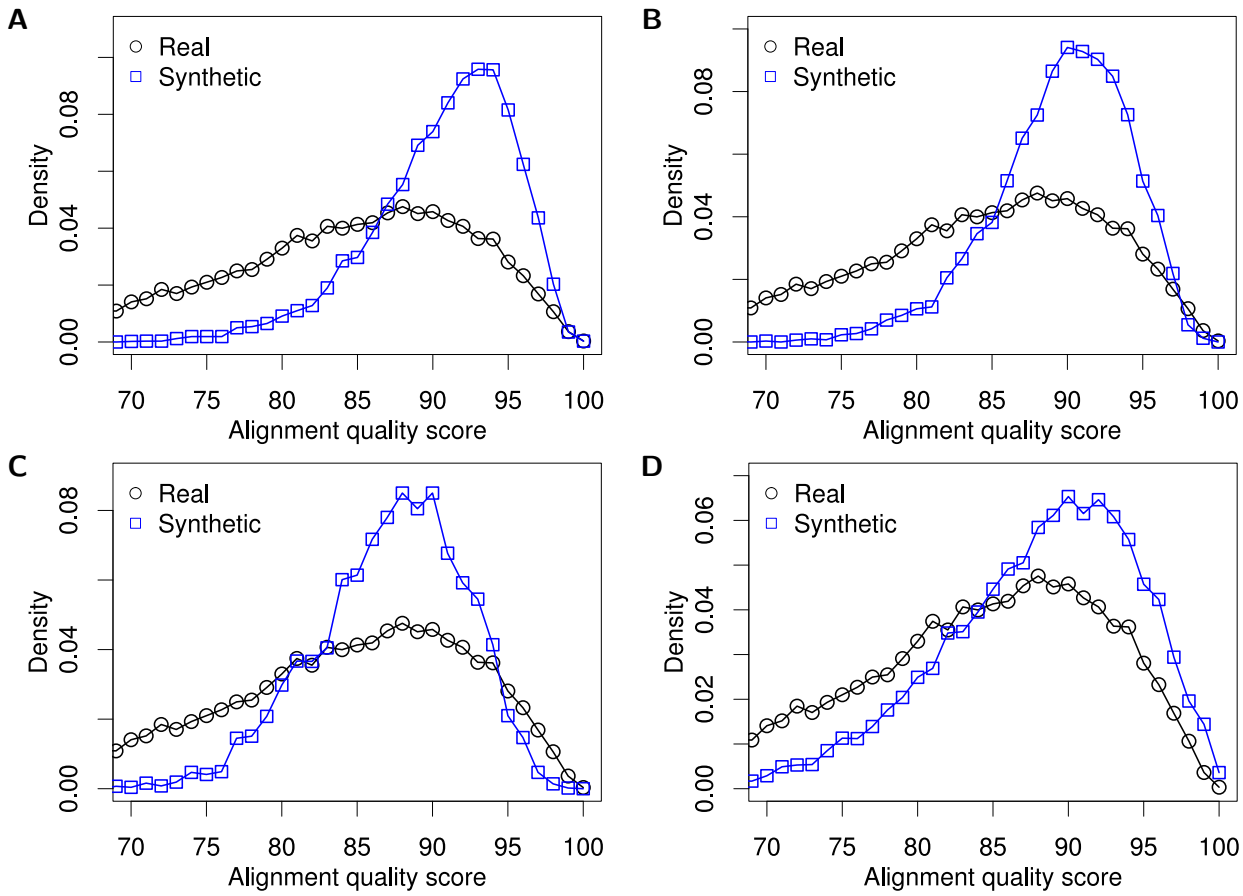


Figure 5.3: Distributions of alignment quality scores (HoT CS) between real and simulated sequences. Synthetic sequences were simulated by (A) a traditional method, (B) using a mixture model of evolutionary rates, (C) using a mixture model of ratios of substitutions to indels, and (D) a novel method that relies on observed genome-wide distributions of its parameters.

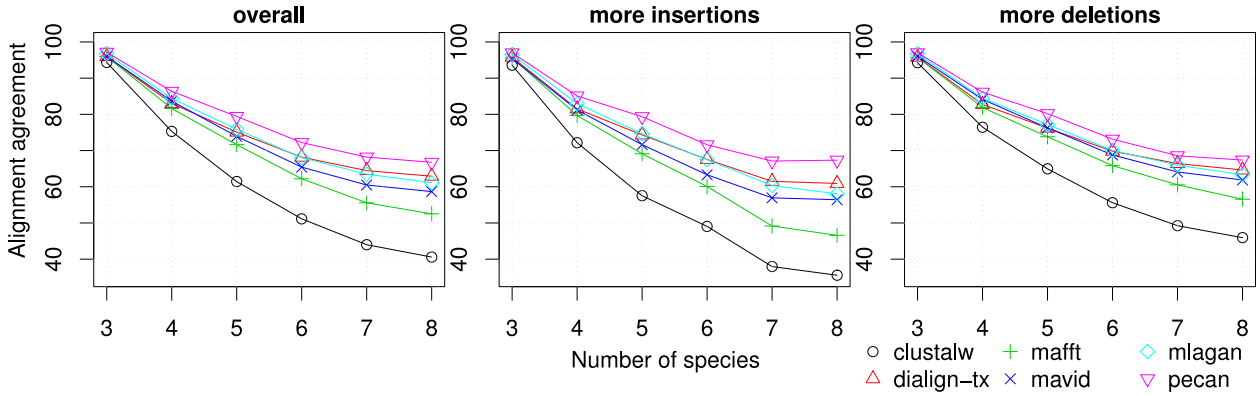


Figure 5.4: Performance of multiple alignment tools compared by alignment agreement. The scores were calculated by using all synthetic data sets (left panel), and by using only data sets where the expected number of insertions is two times more than the number of deletions or vice versa (middle and right panels respectively).

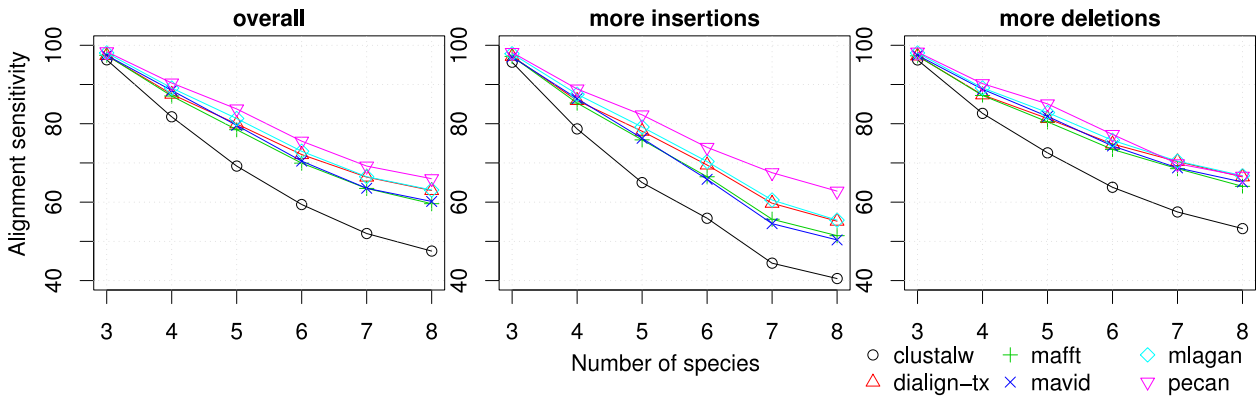


Figure 5.5: Performance of multiple alignment tools compared by alignment sensitivity. The scores were calculated by using all synthetic data sets (left panel), and by using only data sets where the expected number of insertions is two times more than the number of deletions or vice versa (middle and right panels respectively).

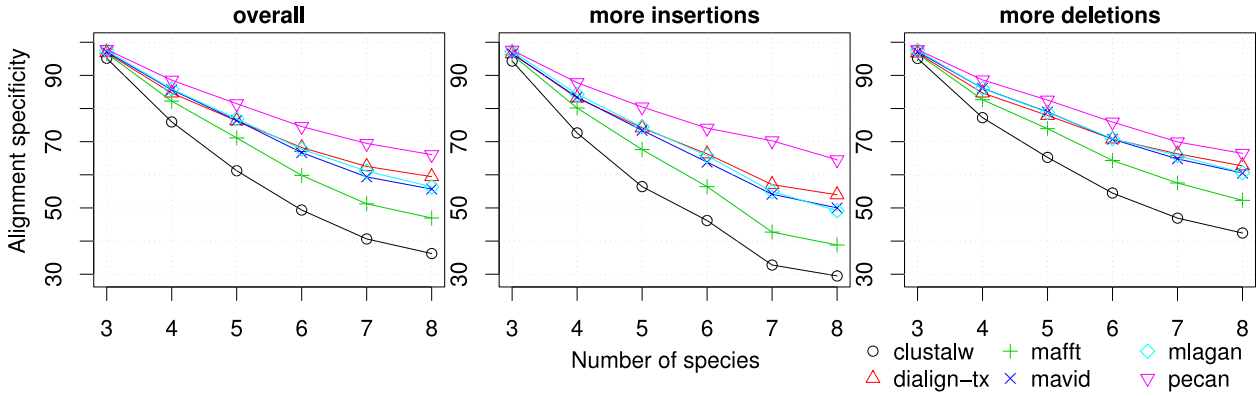


Figure 5.6: Performance of multiple alignment tools compared by alignment specificity. The scores were calculated by using all synthetic data sets (left panel), and by using only data sets where the expected number of insertions is two times more than the number of deletions or vice versa (middle and right panels respectively).

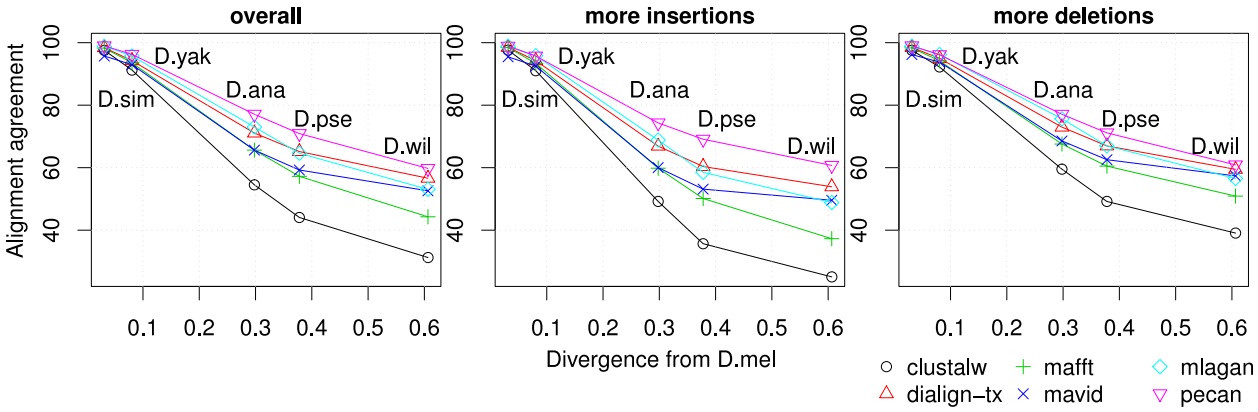


Figure 5.7: Performance of multiple alignment tools compared by alignment agreement of pairs of species. The scores were calculated by using all synthetic data sets (left panel), and by using only data sets where the expected number of insertions is two times more than the number of deletions or vice versa (middle and right panels respectively).

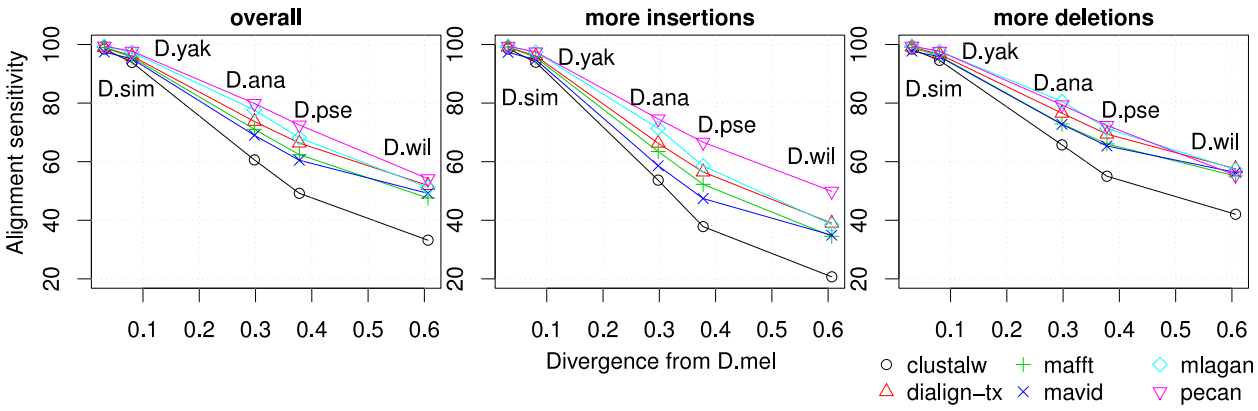


Figure 5.8: Performance of multiple alignment tools compared by alignment sensitivity of pairs of species. The scores were calculated by using all synthetic data sets (left panel), and by using only data sets where the expected number of insertions is two times more than the number of deletions or vice versa (middle and right panels respectively).

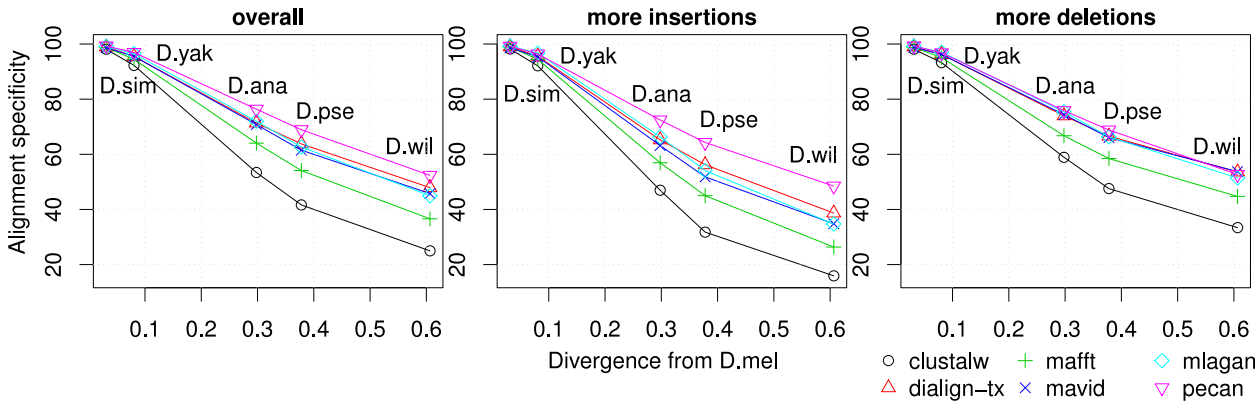


Figure 5.9: Performance of multiple alignment tools compared by alignment specificity of pairs of species. The scores were calculated by using all synthetic data sets (left panel), and by using only data sets where the expected number of insertions is two times more than the number of deletions or vice versa (middle and right panels respectively).

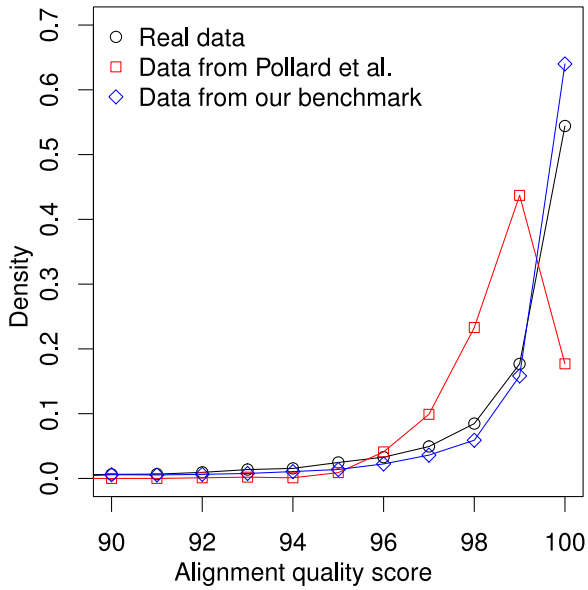


Figure 5.10: Distributions of alignment quality scores of data sets representing *D. melanogaster* - *D. pseudoobscura* pair from real genomes, Pollard et al. [132], and our benchmark. The collected data sets from each of the three sources were aligned by Pecan [124] and then their alignment quality scores were calculated by HoT SPS [86] method.

Table 5.1: Performance of indel annotation tools compared by different measures (ICA, IRA, IAC) on five-species alignments.

	ICA[1]		IRA[2]		IAC[3](sensitivity)		IAC[3](specificity)	
Species	Indelign	sbInfer	Indelign	sbInfer	Indelign	sbInfer	Indelign	sbInfer
Sim	0.06	0.06	1.00	1.01	0.97	0.96	0.99	0.99
Mel	0.04	0.04	1.00	1.01	0.99	0.99	0.99	0.98
Yak	0.06	0.05	1.00	1.01	0.98	0.97	0.97	0.98
Ana	0.08	0.07	1.00	1.00	0.93	0.91	0.93	0.96
Pse	0.24	0.27	1.02	1.03	0.94	0.96	0.79	0.79

[1]Indel Count Agreement (optimal value = 0)
[2]Indel Ratio Agreement (optimal value = 1)
[3]Indel Annotation Coverage (optimal value = 1)

Chapter 6

Probabilistic Model-Based Multiple Sequence Alignment

6.1 Background

Multiple sequence alignment (MSA) is the task of aligning three or more biological sequences, such as DNA, RNA, or proteins. Its purpose is to find homologous regions among the sequences and predict nucleotide or amino acid level relationships among them. MSA is of great importance in downstream analyses of biological sequences, such as a phylogenetic analysis, the identification of patterns of sequence conservation, and protein structure prediction [117, 130]. MSA seeks to optimize an objective function that scores an alignment, and the construction of a biologically realistic objective function is a major challenge.

MSA algorithms can be classified into two broad categories based on how the objective function is defined: score-based alignment and probabilistic alignment. Score-based alignment [116] aims to maximize a similarity score, which is defined based on matching scores for nucleotides or amino acids, and penalties for gaps. However, the parameters of the scoring function cannot be easily estimated, and moreover they are not easy to interpret biologically. On the other hand, probabilistic alignment [8] is based on stochastic modeling of sequence evolution, thus enabling us to use biologically meaningful parameters. In addition, these parameters can be estimated from data by using, for example, the maximum likelihood method.

The first attempt to develop an explicit model of sequence evolution began with the TKF91 model [173]. TKF91 is a continuous-time evolutionary model for insertions, deletions, as well as substitutions of a sequence. Even though many studies have been done to extend the TKF91 model to a more realistic one and to handle multiple (more than two) sequences since then [107, 174], the main problem of this approach is its high computational complexity. A rigorous probabilistic treatment quickly becomes infeasible as the number of sequences increases. This problem can

partially be resolved by using approximate methods, such as Markov chain Monte Carlo (MCMC) [66].

The most widely used approaches to probabilistic MSA are based on expected accuracy to the (unknown) true alignment [37, 124, 145] and a progressive alignment heuristic [46]. Instead of finding the maximum likelihood alignment, the expected accuracy-based alignment algorithms optimize an alignment by maximizing the similarity to the true alignment. The similarity measure is usually defined as the expected number of correctly aligned residues, which can be easily estimated by using posterior pairwise alignment probabilities. To reduce computational cost, an MSA is usually constructed by a progressive alignment heuristic, which first builds a guide tree based on the similarities between sequences and then grows the MSA by repeatedly aligning a pair of sequences or alignments (profiles). However, the progressive alignment heuristic has a serious drawback: it propagates errors made in early stages to later stages because of its large alignment step (a whole sequence). Many progressive alignment-based algorithms use an "iterative refinement" step to address the problem but this is not a complete solution, especially for complex errors. As an attempt to reduce the chance of the early errors, a consistency transformation technique [37] has also been developed, which updates the posterior pairwise alignment probabilities by incorporating consistency with other sequences.

One major problem of the expected accuracy measure is that it is asymmetric, in that it only considers aligned residues, and therefore it may perform poorly when there are many unaligned residues in the true alignment. To address this problem, a new similarity measure, called "Alignment Metric Accuracy" (AMA), has been developed [149], which is a symmetric measure that can take into account unaligned residues as well as the aligned ones and provide us a mechanism to control the trade-off between them. The new similarity measure has been successfully applied to recent MSA algorithms [14, 148]. Instead of using the progressive alignment scheme, these alignment algorithms are based on a graph-based scheme and greedily construct an MSA by incrementally creating alignment columns whose contribution to increasing an alignment accuracy is the highest at each step.

However, all of the above similarity measures approximate the similarity between two MSAs by using a sum-of-pairs (SP) scheme. The basic idea of the SP approach is to sum the contributions

of every pair of residues. Although it is a good way of reducing the complexity of calculating the similarity score, the simple summation may not accommodate the relationships among all sequences. Therefore, we need a more principled way that can directly take into account the dependency of multiple sequences.

In this chapter, we present a novel probabilistic alignment algorithm for multiple sequences that is able to address the aforementioned problems. The algorithm is based on (i) an evolutionary model for three sequences, (ii) a direct estimation of the joint probability of each alignment column instead of using the sum-of-pairs (SP) scheme, and (iii) the use of a sequence annealing [148], which is a kind of graph-based alignment algorithm, as an alternative to the progressive alignment heuristic. The new alignment algorithm is compared to the most recent and accurate MSA tools, by using the realistic benchmarks developed in Chapter 5. We find that the performance of the new algorithm is overall comparable or superior to existing MSA tools.

6.2 Alignment Algorithm

6.2.1 Overview

Our "Probabilistic Evolutionary model-based Multiple Alignment" (PEMA) algorithm is a triple-HMM-based sequence annealing algorithm. From the input multiple sequences and a phylogenetic tree with branch lengths, PEMA first performs the three-way analysis of input sequences to obtain posterior probabilities that three residues from different sequences are aligned. The posterior triplewise alignment probabilities are computed by using the standard Forward-Backward algorithms [138] of the triple-HMM that represents the evolution of three sequences. The standard computation of the posterior probabilities has very high time and memory requirements. PEMA addresses this problem by reducing the number of triplets of residues that are compared, based on two-way comparisons of sequences. Once the computation of the posterior triplewise alignment probabilities is done, PEMA constructs an MSA by using a non-progressive alignment algorithm, called "sequence annealing" [148], which builds an MSA by incrementally making alignment columns.

We implemented our algorithm by modifying the implementation of the FSA algorithm [14], which seeks to maximize the expected accuracy of an MSA by using the sequence annealing algo-

rithm. While FSA uses a sum-of-pairs scheme to define the accuracy of an MSA, PEMA directly estimates the joint probabilities of multiple residues in alignment columns by using an iterative approximation method.

6.2.2 Model for the Evolution of Three Sequences

The evolution of sequences is generally assumed to follow a binary tree, and the tree for three sequences can be represented as in Figure 6.1A. There is a common ancestor of the close two sequences X and Y, and it is connected to the sequence Z via a root sequence. The modeling of the evolution shown in Figure 6.1A, however, needs to consider two hidden sequences at the two ancestral nodes. A simple yet reasonable way is to assume the sequence Z as the root of the tree and only keep one internal node that connects the three sequences (Figure 6.1B). This is motivated by the fact that the root can be placed anywhere between the sequence Z and the common ancestor of the sequences X and Y if the underlying Markov process that generates the sequences is reversible [44].

An evolutionary model describes the probabilities of three mutation events (substitutions, insertions, and deletions) along each branch of the tree shown in Figure 6.1B, and it can be efficiently represented by a hidden Markov model (HMM) structure. However, the explicit model for the three sequences based on the tree in Figure 6.1B demands a very large number of different states in the HMM, resulting in intractable computational complexity. For example, even if we use a simple three-state HMM, such as the one in Figure 6.3, for modeling the evolution of a sequence on each branch in the tree in Figure 6.1B, we need 15 different states in the extended HMM for three sequences [48, 66]. Our preliminary experiments have shown that such a model is not computationally tractable to align three sequences of length a few Kbp (data not shown). We therefore assumed that the three sequences follow a star topology (Figure 6.1C) that is tractable and can represent the dependency among three sequences, although it is not an entirely realistic model.

The triple-HMM with seven states in Figure 6.2 represents the emission of three sequences based on our evolutionary model, as a simple extension of a pair-HMM for three sequences. Specifically, in the state "ZXY", three nucleotides for each sequence are emitted. The states "ZX−", "Z−Y", and "−XY" represent the case where only two of the three sequences have emitted nucleotides.

The remaining states "Z−−", "−X−", and "−−Y" are responsible for the case where a nucleotide is emitted for only one sequence. Table 6.1 shows the transition probabilities between the states in the triple-HMM.

6.2.3 Computation of the Posterior Triplewise Alignment Probability

Suppose we are given N sequences, $S = \{X_1, \ldots, X_N\}$. For every triplet of sequences $X, Y, Z \in S$ and all $i \in \{1, \ldots, |X|\}$, $j \in \{1, \ldots, |Y|\}$, $k \in \{1, \ldots, |Z|\}$, PEMA uses the seven-state triple-HMM shown in Figure 6.2 to compute the probability, $P(x_i \sim y_j \sim z_k | X, Y, Z)$, that nucleotides x_i, y_j, and z_k are aligned. PEMA first compares two pairs of sequences Z and X, and Z and Y to compute the posterior probabilities that two nucleotides at each position of two different sequences are aligned. This is done by using the simple pair-HMM (Figure 6.3) with three states. We used the standard Forward-Backward algorithms [138] on the pair-HMM to compute the posterior probabilities, which were used to reduce the search space of the comparisons of three sequences at later stage.

After obtaining the posterior pairwise alignment probabilities, PEMA computes the posterior triplewise alignment probability, $P(x_i \sim y_j \sim z_k | X, Y, Z)$, by using the standard Forward-Backward algorithms [138]. The Forward-Backward algorithms are dynamic programming algorithms that compute the probability of a particular observation of sequences by filling the dynamic programming (DP) table. The time and memory complexity of such algorithms are $O(N_s^2 L^N)$ and $O(N_s L^N)$, respectively, where N_s is the number of states in an HMM, L is the average length of sequences, and N is the number of sequences. These complexities, however, can be dramatically reduced by observing that only a small region of the DP table contributes to the total likelihood. We can find such a high contributing region in the DP table for the three sequences by using the posterior pairwise alignment probabilities mentioned above. Specifically, if the posterior probabilities $P(x_i \sim z_k | X, Z)$ and $P(y_j \sim z_k | Y, Z)$ are greater than a threshold, then the cell in the DP table corresponding to (x_i, y_j, z_k) may belong to the high contributing region and the forward-backward probabilities of the cell are computed.

6.2.4 Estimation of the Joint Probability of an Alignment Column

Probabilistic alignment algorithms based on the expected accuracy [14, 37, 124, 148] try to find the most accurate alignment instead of the maximum likelihood one. For example, in the case of two sequences X and Y, the expected accuracy of an alignment A^* can be simply defined as:

$$\frac{1}{\min\{|X|,|Y|\}} \sum_{x_i \sim y_j \in A^*} P(x_i \sim y_j \in A|X,Y)$$

where A is the true alignment, and the posterior probability computed from the pair-HMM is commonly used as a good estimate of the probability $P(x_i \sim y_j \in A|X,Y)$. This method is extended to multiple sequences by summing the posterior probabilities over all pairs of sequences. As an alternative to the sum-of-pairs method [169], we can directly estimate the posterior probability for *more than two sequences* in an attempt to define a more accurate similarity measure to the true alignment.

As described in the previous section, the posterior probabilities for three sequences can be computed based on the triple-HMM by using the standard algorithms. Due to the high computational complexity, however, the direct extension of the HMM to more than three sequences is not tractable. Therefore, we need a method that can estimate the posterior probabilities for more than three sequences by using available information, the posterior triplewise alignment probabilities in our case. For this purpose, we converted the problem to a problem of estimating the cell probabilities of a contingency table for which the marginal probabilities are known [69]. Here, the cell probabilities and the marginals correspond to the joint probabilities of alignment columns with more than three residues and the posterior triplewise alignment probabilities, respectively.

Suppose there is an $r \times c$ contingency table, and the marginal probabilities $p_{i\bullet}$ and $p_{\bullet j}$, which are defined by the following equations for each row and column, are known.

$$p_{i\bullet} = \sum_{j=1}^{c} p_{ij} \quad (i = 1, 2, \ldots, r) \qquad p_{\bullet j} = \sum_{i=1}^{r} p_{ij} \quad (j = 1, 2, \ldots, c)$$

Then, the probability p_{ij} can be estimated based on the principle of maximum entropy [72] as a way to find a probability distribution which best describes the current state without introducing

any subjective additional information [47, 51, 69]. The solutions are

$$p^*_{ij} = a_i b_j \pi_{ij} \qquad p_{i\bullet} = a_i \sum_j b_j \pi_{ij} \qquad p_{\bullet j} = b_j \sum_i a_i \pi_{ij}$$

where $\Sigma_i \Sigma_j \pi_{ij} = 1$ and the estimator p^*_{ij} can be computed by iterations that are defined as [69, 70]

$$\begin{aligned} p_{ij}^{(2n+1)} &= \frac{p_{i\bullet}}{p_{i\bullet}^{(2n)}} p_{ij}^{(2n)}, \\ p_{ij}^{(2n+2)} &= \frac{p_{\bullet j}}{p_{\bullet j}^{(2n+1)}} p_{ij}^{(2n+1)}, \quad n = 0, 1, \ldots, \quad p_{ij}^{(0)} = \pi_{ij} \end{aligned}$$

Figure 6.4 shows how the iterations work on a simple 2×2 contingency table. At each iteration, the method updates the cell probabilities twice by (i) the ratio between the given "row" marginals and those estimated at the previous step and (ii) the ratio between the given "column" marginals and those estimated at the previous step. The iteration stops when there is no more change of the cell probabilities or the number of iterations exceeds a given threshold. Ireland and Kullback [69] proved the convergence of this iterative method.

This method is not limited to the two-way contingency table and can work with multi-way and mixed marginals. In the context of estimating the joint probability, $P(x_{1l_1}, x_{2l_2}, \ldots, x_{Kl_K})$, that aligns K residues from K sequences $X_1, \ldots, X_K$, we need to consider a $|X_1| \times \ldots \times |X_K|$ contingency table where $|X_i|$ is the length of the sequence X_i. (Here, we assumed that the joint probability has the same semantics as the posterior alignment probability $P(x_{1l_1} \sim \cdots \sim x_{Kl_K})$.) However, the high dimensionality with the large number of possible values of each dimension makes the method difficult to be applied to our problem. In addition, not all of the posterior probabilities of the different positions of the sequences need to be computed because only small portion of them contribute to the total likelihood, implying that the rest of them may have very small probabilities close to 0. Motivated by this observation, we reduced the possible value of the dimension k ($k \in \{1, \ldots, K\}$) to two by considering only two categories: (i) the l_k^{th} residue of a sequence X_k, denoted by x_{kl_k} and (ii) all other residues of the sequence X_k, denoted by $\overline{x_{kl_k}}$, and computed the probabilities on-the-fly as needed. As a result, we obtained a $2 \times \ldots \times 2$ contingency

table where marginal probabilities for each triplet of dimensions are provided as follows.

$$
\begin{aligned}
P(\overline{x_{il_i}}, x_{jl_j}, x_{kl_k}) &= P(x_{jl_j}, x_{kl_k}) - P(x_{il_i}, x_{jl_j}, x_{kl_k}) \\
P(x_{il_i}, \overline{x_{jl_j}}, x_{kl_k}) &= P(x_{il_i}, x_{kl_k}) - P(x_{il_i}, x_{jl_j}, x_{kl_k}) \\
P(x_{il_i}, x_{jl_j}, \overline{x_{kl_k}}) &= P(x_{il_i}, x_{jl_j}) - P(x_{il_i}, x_{jl_j}, x_{kl_k}) \\
P(\overline{x_{il_i}}, \overline{x_{jl_j}}, x_{kl_k}) &= P(x_{kl_k}) - P(x_{il_i}, x_{kl_k}) - P(x_{jl_j}, x_{kl_k}) + P(x_{il_i}, x_{jl_j}, x_{kl_k}) \\
P(\overline{x_{il_i}}, x_{jl_j}, \overline{x_{kl_k}}) &= P(x_{jl_j}) - P(x_{il_i}, x_{jl_j}) - P(x_{jl_j}, x_{kl_k}) + P(x_{il_i}, x_{jl_j}, x_{kl_k}) \\
P(x_{il_i}, \overline{x_{jl_j}}, \overline{x_{kl_k}}) &= P(x_{il_i}) - P(x_{il_i}, x_{jl_j}) - P(x_{il_i}, x_{kl_k}) + P(x_{il_i}, x_{jl_j}, x_{kl_k}) \\
P(\overline{x_{il_i}}, \overline{x_{jl_j}}, \overline{x_{kl_k}}) &= \sum_{l_{i'}} \sum_{l_{j'}} \sum_{l_{k'}} P(x_{il_{i'}}, x_{jl_{j'}}, x_{kl_{k'}}) - P(x_{il_i}, x_{jl_j}, x_{kl_k})
\end{aligned}
$$

where $P(x_{il_i}, x_{jl_j}, x_{kl_k})$ is equal to the posterior probability $P(x_{il_i} \sim x_{jl_j} \sim x_{kl_k})$ that is computed by the triple-HMM, and $P(x_{il_i}, y_{jl_j})$ and $P(x_{il_i})$ are computed from $P(x_{il_i}, x_{jl_j}, x_{kl_k})$ by marginalization.

For example, suppose $K = 5$. Together with the assumption that the initial cell probabilities are uniform and thus equal to $1/2^5$, we can now estimate the cell probabilities by the following iterations.

$$
\begin{aligned}
p_{ijklm}^{(10n+1)} &= \frac{p_{ijk\bullet\bullet}}{p_{ijk\bullet\bullet}^{(10n)}} p_{ijklm}^{(10n)}, \\
p_{ijklm}^{(10n+2)} &= \frac{p_{ij\bullet l\bullet}}{p_{ij\bullet l\bullet}^{(10n+1)}} p_{ijklm}^{(10n+1)}, \\
p_{ijklm}^{(10n+3)} &= \frac{p_{ij\bullet\bullet m}}{p_{ij\bullet\bullet m}^{(10n+2)}} p_{ijklm}^{(10n+2)}, \\
&\cdots \\
&\text{where } \; n = 0, 1, \ldots, \quad p_{ijklm}^{(0)} = 1/2^5, \quad i, j, k, l, m \in \{1, 2\}
\end{aligned}
$$

Here, the cell probability p_{ijklm} corresponds to the joint probability $P((x_{1l_1})^{2-i} \cdot (\overline{x_{1l_1}})^{i-1}, \ldots, (x_{5l_5})^{2-m} \cdot (\overline{x_{5l_5}})^{m-1})$.

To evaluate the iterative method, we simulated the contingency table and computed marginal probabilities of each triplet of dimensions. Then, we used the iterative method to re-estimate the cell probabilities by only using the marginal probabilities. We compared the estimated cell

probabilities with the simulated ones by using the relative entropy, which is a measure of the difference between two probability distributions. To capture the non-uniformity of the posterior probabilities for aligning residues, we used the Dirichlet distribution to sample the cell probabilities. We repeated the evaluation for the dimensions from four to eight. Table 6.2 shows that the relative entropy values for the range of the different number of dimensions are very small and therefore the iterative method works reasonably well.

6.2.5 FSA Algorithm

The FSA program [14] aims to find the alignment with the minimum expected distance to the true alignment (A). In the case of two sequences X and Y, the optimal alignment is therefore

$$A_{optimal} = \underset{A^*}{\operatorname{argmin}}\, \mathbb{E}[d(A^*, A)]_{P(A|X,Y)} \tag{6.1}$$

The distance $d(A^*, A)$ between two alignments is defined as the number of residues in the two sequences X and Y for which they make different homology statements. The homology statement for the i^{th} residue of X (x_i) has two forms: either $x_i \sim y_j$ (x_i is homologous to the j^{th} residue of Y) or $x_i \sim -$ (x_i is not homologous to any position in Y). The distance between two alignments $d(A^*, A)$ is equal to $|X| + |Y| - Sim(A^*, A)$, where $Sim(A^*, A)$ is a similarity measure defined as the number of residues for which A^* and A make identical homology statements:

$$\begin{aligned} Sim(A^*, A) &= 2 \sum_{i,j:x_i \sim y_j \in A^*} \mathbf{1}\{x_i \sim y_j \in A\} \\ &\quad + \sum_{i:x_i \sim - \in A^*} \mathbf{1}\{x_i \sim - \in A\} + \sum_{j:- \sim y_j \in A^*} \mathbf{1}\{- \sim y_j \in A\} \end{aligned}$$

Therefore, finding the optimal alignment with the minimum expected distance is the same as finding the alignment with the maximum expected similarity, that is

$$\begin{aligned} A_{optimal} &= \underset{A^*}{\operatorname{argmin}}\, \mathbb{E}[d(A^*, A)]_{P(A|X,Y)} \\ &= \underset{A^*}{\operatorname{argmin}}\, \mathbb{E}[|X| + |Y| - Sim(A^*, A)]_{P(A|X,Y)} \\ &= \underset{A^*}{\operatorname{argmax}}\, \mathbb{E}[Sim(A^*, A)]_{P(A|X,Y)} \end{aligned} \tag{6.2}$$

By using the posterior pairwise alignment probabilities $P(A|X,Y)$ computed based on the statistical model (a pair-HMM) in FSA, Equation 6.2 is further reduced to

$$\begin{aligned} A_{optimal} &= \underset{A^*}{\operatorname{argmax}}\, \mathbb{E}[Sim(A^*, A)]_{P(A|X,Y)} \\ &= \underset{A^*}{\operatorname{argmax}} \sum_{A} [P(A|X,Y) Sim(A^*, A)] \\ &= \underset{A^*}{\operatorname{argmax}} \left[2 \sum_{i,j:x_i \sim y_j \in A^*} P(x_i \sim y_j | X, Y) \right. \\ &\quad \left. + \sum_{i:x_i \sim - \in A^*} P(x_i \sim - | X, Y) + \sum_{j:- \sim y_j \in A^*} P(- \sim y_j | X, Y) \right] \end{aligned} \tag{6.3}$$

To control the sensitivity/specificity trade-off, FSA program introduces a gap factor "gf" into the objective function in Equation 6.3:

$$\begin{aligned} A_{optimal} &= \underset{A^*}{\operatorname{argmax}} \left[2 \sum_{i,j:x_i \sim y_j \in A^*} P(x_i \sim y_j | X, Y) \right. \\ &\quad \left. + \mathrm{gf} \left(\sum_{i:x_i \sim - \in A^*} P(x_i \sim - | X, Y) + \sum_{j:- \sim y_j \in A^*} P(- \sim y_j | X, Y) \right) \right] \end{aligned}$$

A lower gap factor produces a more sensitive alignment and a higher gap factor generates a more specific alignment. FSA uses gf=1 by default. When gf=0, the most sensitive alignments are produced.

The objective function can be extended for more than two sequences by taking sum-of-pairs over all sequences. Given N sequences $X_1, \ldots, X_N$ related by a phylogenetic tree T, the optimal alignment by the generalized objective function of Equation 6.1 is

$$\begin{aligned} A_{optimal} &= \underset{A^*}{\operatorname{argmin}}\, \mathbb{E}[d(A^*, A|T)]_{P(A|X_1,\ldots,X_N,T)} \\ &= \underset{A^*}{\operatorname{argmin}} \sum_{A} P(A|X_1, \ldots, X_N, T) d(A^*, A|T) \\ &= \underset{A^*}{\operatorname{argmin}} \left[\frac{1}{\binom{N}{2}} \sum_{i,j} \sum_{A_{i,j}} \sum_{A|A_{i,j}} P(A|X_1, \ldots, X_N, T) d(A^*, A|T) \right] \end{aligned} \tag{6.4}$$

This objective function is further reduced by applying two restrictions: (i) the distance $d(A^*, A|T)$

is a weighted sum of pairwise distances,

$$d(A^*, A|T) = \sum_{i,j} w_{ij}(T) d(A^*_{ij}, A_{ij}) \tag{6.5}$$

and (ii) the full probabilistic model is approximated as a pairwise model,

$$\sum_{A|A_{ij}} P(A|X_1, \ldots, X_N, T) = P(A_{ij}|X_i, X_j) \tag{6.6}$$

The weight $w_{ij}(T)$ can be used to account for a known phylogeny. FSA uses $w_{ij}(T) = 1$ for all i, j, T, representing a phylogeny-free approach. These restrictions result in the following sum-of-pairs objective function used by FSA program.

$$\mathbb{E}[d(A^*, A|T)]_{P(A|X_1,\ldots,X_N,T)} = \sum_{i,j} w_{i,j}(T) \sum_{A_{i,j}} d(A^*_{ij}, A_{ij}) P(A_{ij}|X_i, X_j) \tag{6.7}$$

Sequence annealing [14, 148] is a greedy algorithm that aligns multiple sequences by repeatedly merging two existing columns in an alignment, such that their contribution (weight) to the increase of the similarity to the true alignment is highest, until a stopping criteria is reached. Specifically, the sequence annealing process begins with the null alignment, where all sequences are unaligned and each alignment column contains only one residue. At each iteration, it maintains a sorted list of all pairs of columns in a current alignment according to their weights computed by a weighting function, which is motivated by the objective function and defined as

$$\begin{aligned} w(col_1, col_2) = {} & 2 \sum_{x_i : X_i \in col_1} \sum_{y_j : X_j \in col_2} P(x_i \sim y_j | X, Y) \\ & \Big/ \left(\sum_{x_i : X \in col_1} \sum_{y_j : Y \in col_2} [P(x_i \sim -|X, Y) + P(- \sim y_j | X, Y)] \right) \end{aligned}$$

Then it merges two columns whose weight is the highest and whose merging is not inconsistent with previously merged columns. As the merge process goes on, the weights are efficiently re-calculated as needed to reflect newly aligned columns (see [14] for more technical details). FSA views an alignment as a Directed Acyclic Graph (DAG), whose nodes correspond to the alignment columns

and edges represent the order of the columns. Therefore, merging two columns is equivalent to merging two nodes and maintaining alignment consistency is equivalent to keeping the graph acyclic. A global alignment can be produced by choosing a linear extension of the graph by a topological ordering.

6.2.6 PEMA Algorithm

The main approximation of the objective function of FSA is the use of sum-of-pairs over all sequences (Equation 6.7). The sum-of-pairs is a meaningful approach as formalized in Equation 6.4. However, the common problem of the sum-of-pairs approach is its inability to fully account for dependency among multiple (more than two) sequences that are related to each other as products of evolution. A direct way to address this problem is to use the full probabilistic model for multiple sequences and compute the posterior alignment probabilities conditioned on multiple sequences. However, it is computationally intractable and this is the reason why the pairwise approximation is used in the second restriction (Equation 6.6). This can be partially resolved if we can estimate the posterior alignment probabilities by using a reduced probabilistic model that can represent more than two sequences and yet is computationally tractable.

To this end, we started by relaxing the two restrictions imposed on Equations 6.5 and 6.6. Instead of only using the sum-of-pairs approach, we redefined the similarity measure $Sim(A^*, A)$ between two MSAs by directly comparing alignment columns and treating the homology statement of being aligned to any position (non-gaps) differently from that of being aligned to gaps. In other words, the new similarity measure $newSim(A^*, A)$ is the sum of the following two numbers.

- The weighted number of columns for which A^* and A make identical homology statements
- Over all alignments of two sequences X_i and X_j, the sum of the numbers of residues being aligned to gaps for which A^*_{ij} and A_{ij} make identical homology statements

A rigoruos form of the homology statement for the i^{th} column (col_i) with n residues from n sequences is $x_1 \sim ... \sim x_n$, where x_j ($j \in \{1, \cdots, n\}$) is a residue in that column. However, for computational tractability, we imposed one restriction that A^* and A make identical homology statements for the column col_i in A^* if there is a column in A that contains all residues in the column col_i. The weight

$w(col_i)$ of the column col_i is defined as follows and this results in balanced numbers of residues between those aligned to gaps and non-gaps in the new similarity measure.

$$w(col_i) = 2 \times \binom{|col_i|}{2} = |col_i| \times (|col_i| - 1)$$

where $|col_i|$ is the number of residues in that column. The similarity for residues being aligned to gaps (the latter) is the same as the one defined in FSA by using the sum-of-pairs approach.

Therefore, the new similarity measure $newSim(A^*, A)$ can be defined as

$$\begin{aligned} newSim(A^*, A) &= \sum_{col_i \in A^*} w(col_i) \cdot \mathbf{1}\{col_i \in' A\} \\ &\quad + \sum_{a,b} \left[\sum_{i: x_i \sim - \in A^*_{ab}} \mathbf{1}\{x_i \sim - \in A_{ab}\} + \sum_{j: - \sim y_j \in A^*_{ab}} \mathbf{1}\{- \sim y_j \in A_{ab}\} \right] \end{aligned}$$

As mentioned above, the column col_i is said to belong to the alignment A ($col_i \in' A$) if there is a column in the alignment A that contains all residues in the column col_i. Based on the new similarity measure, the optimal alignment in PEMA is

$$\begin{aligned} A_{optimal} &= \underset{A^*}{\operatorname{argmax}}\, \mathbb{E}[newSim(A^*, A)]_{P(A|X_1,\ldots,X_N,T)} \\ &= \underset{A^*}{\operatorname{argmax}} \sum_{A} [P(A|X_1, \ldots, X_N, T) newSim(A^*, A)] \\ &= \underset{A^*}{\operatorname{argmax}} \left[\sum_{col_i \in A^*} w(col_i) P(col_i | X_1, \ldots, X_N, T) \right. \\ &\quad \left. + \sum_{a,b} \left(\sum_{i: x_i \sim - \in A^*_{ab}} P(x_i \sim - | X_a, X_b) + \sum_{j: - \sim y_j \in A^*_{ab}} P(- \sim y_j | X_a, X_b) \right) \right] \end{aligned}$$

As in the FSA program, we used a gap factor "gf" to control the sensitivity/specificity trade-off.

Based on the above objective function, the new weighting function for the sequence annealing

in PEMA is defined as

$$w(col_1, col_2) = w(col_{12}) P(col_{12}|X_1, \ldots, X_N, T) \Bigg/ \left(\sum_{x_i : X \in col_1} \sum_{y_j : Y \in col_2} [P(x_i \sim -|X, Y) + P(- \sim y_j|X, Y)] \right)$$

where col_{12} is a new column generated by merging the two columns col_1 and col_2. The column probability $P(col_i|X_1, \ldots, X_N, T)$ is the joint probability of aligned residues in the column col_{12} and it is approximated by the method presented in Section 6.2.4.

6.3 Comparison to Other Multiple Alignment Tools

We compared the alignments of PEMA on the benchmarks for the alignments of *Drosophila* non-coding sequences developed in Chapter 5 to those of two multiple alignment tools for DNA sequences: Pecan [124] that is based on a progressive alignment heuristic and chosen as the best tool in Chapter 5, and FSA [14] that is based on a sequence annealing heuristic. They are both a probabilistic alignment tool that uses posterior pairwise alignment probabilities, which are obtained from a pair-HMM, and attempts to find the alignment with the optimal expected accuracy. The choice of these two programs enables us to compare the sequence annealing to the progressive alignment that is the most widely used alignment heuristic.

We used the following three commonly used evaluation measures [11, 14] introduced in Chapter 5: (i) *alignment agreement*, which is the fraction of aligned base pairs (or bases aligned to gaps) in the predicted alignment that agree with the true alignment, (ii) *alignment sensitivity*, which is the fraction of aligned base pairs of the true alignment that agree with the predicted alignment, and (iii) *alignment specificity*, which is the fraction of aligned base pairs of the predicted alignment that agree with the true alignment. Whereas the alignment agreement score considers aligned base pairs as well as bases aligned to gaps, the sensitivity and specificity scores are calculated *only* from aligned base pairs.

The results of our evaluations are shown in Figures 6.5 and 6.6 on benchmarks with five and eight species, respectively. We performed the same evaluations by limiting ourselves to those data sets (in the benchmark) that had an excess of deletions over insertions, and separately to those data

sets with an excess of insertions (Figures 6.5 and 6.6; middle and right bars). The FSA and PEMA were run with two different options, default and maximum sensitivity, that control the sensitivity and specificity of an alignment.

In terms of the alignment agreement (Figures 6.5A and 6.6A), PEMA was found to be superior to FSA with both options. Pecan outperformed the other tools on the five species benchmark. In contrast, the agreement score of Pecan became very similar to that of PEMA with the default option on eight species benchmark. As the number of species increases from five to eight, both PEMA and FSA produced higher-agreement alignments with the default option than the maximum sensitivity option. This is probably due to the increased number of residues aligned to gaps because the total divergence of the eight species benchmark is larger than that of the five species benchmark.

In the case of the alignment sensitivity (Figures 6.5B and 6.6B), the maximum sensitivity option did have an impact on improving the sensitivity for both FSA and PEMA. Similar to the trend observed from the alignment agreement measure, the performance of PEMA was better than FSA, and PEMA with the maximum sensitivity option achieved comparable performance to Pecan on the eight species benchmark although Pecan performed slightly better than PEMA on the five species benchmark.

On the other hand, PEMA with the default option produced the most specific alignments on both the five and eight species benchmarks (Figures 6.5C and 6.6C). The specificity score of Pecan was substantially lower than those of PEMA and FSA with the default option. This is not surprising because the similarity measure of Pecan does not consider unaligned bases, which is likely to generate many over-aligned base pairs.

These results indicate that (i) the sequence annealing by using the joint probability of an alignment column (the approach of PEMA) is better at improving both sensitivity and specificity of an alignment than the sequence annealing with the sum-of-pairs method by using the posterior pairwise alignment probabilities (the approach of FSA), (ii) the similarity measure that considers unaligned residues as well as aligned ones is superior to the measure that only relies on aligned residues for producing a more specific alignment, and (iii) in most cases with different evaluation measures and different alignment tools, the performance on the data sets with an excess of deletions than insertions was better than that on the opposite configuration of data sets.

6.4 Comparison to Variants of Other Multiple Alignment Tools

We next compared PEMA with two variants of other tools, FSAPEMA and ProbConsPEMA. These are modified versions of FSA and ProbCons [37] tools, respectively, that take advantage of the posterior triplewise alignment probabilities to estimate the posterior pairwise alignment probabilities by marginalization. ProbCons is a probabilistic multiple alignment tool that is based on a progressive alignment heuristic and the expected accuracy measure as the tool Pecan. To increase the accuracy of an alignment, ProbCons applies a consistency transformation, which updates the posterior pairwise alignment probabilities by incorporating the consistency to other sequences. However, this feature was turned off in this evaluation because ProbConsPEMA re-estimates the posterior pairwise alignment probabilities from triplewise probabilities, whose effect is similar to the transformation in the sense of considering additional information from other sequences. We wanted to see the effect of the posterior triplewise alignment probabilities, not the consistency transformation technique. We chose to use ProbCons, not Pecan, because the source of Pecan was not available.

Figures 6.7 and 6.8 show the evaluation results on the same benchmarks used in the previous section, with five and eight species, respectively. The agreement score of ProbConsPEMA was the highest, which was followed by PEMA, when evaluated on the five species benchmark (Figure 6.7A). However, PEMA with the default option was superior to all other tools on the eight species benchmark (Figure 6.8A).

When comparing the sensitivity scores (Figures 6.7B and 6.8B), ProbConsPEMA performed the best on both the five and eight species benchmarks. The scores of PEMA and FSAPEMA with the maximum sensitivity option were very similar to each other on the five species benchmark (Figures 6.7B), whereas the score of FSAPEMA was slightly higher than that of PEMA on the eight species benchmark (Figures 6.8B).

On the other hand, the alignment specificity score showed a little different trend (Figures 6.7C and 6.8C). The specificity score of FSAPEMA was exceptionally high (about 94%) and the score was almost unchanged with the increased number of species from five to eight. This may result from the large decrease of aligned bases as being indicated by the relatively low sensitivity scores (Figures 6.7B and 6.8B). ProbConsPEMA generated the worst specific alignments. This is again

due to the fact that ProbConsPEMA does not care about unaligned bases when assessing the expected accuracy of an alignment, which is similar to Pecan.

Based on these observations, we can see that the combination of the progressive alignment and the posterior pairwise alignment probabilities obtained by marginalizing the posterior triplewise alignment probabilities (the approach of ProbConsPEMA) is a simple yet competitive option to increase the agreement and sensitivity of alignments. This is supported by the observation that the scores of ProbConsPEMA were highest or comparable to other tools in terms of the alignment agreement and sensitivity on the five and eight species benchmarks. In addition, the performance of PEMA is not just the result of the use of a complex model (the model of three sequences instead of two sequences) because when comparing the agreement and sensitivity scores, the performance of FSAPEMA, which is also based on the model of three sequences, was comparable to or worse than that of PEMA, suggesting that the joint probability of alignment column is a promising alternative to the sum-of-pairs scheme. Finally, the similarity measure that considers both aligned and unaligned residues plays a critical role in increasing the specificity of alignments because the alignment specificity scores of PEMA and FSAPEMA were better than those of ProbConsPEMA that is only based on an aligned residue-based similarity measure.

6.5 Discussion

Multiple sequence alignment has been extensively studied over the past several decades due to its importance as a preprocessing step for downstream analyses of biological sequences. As a result, many alignment tools have been developed, and among them, probabilistic alignment algorithms based on the model of sequence evolution are the most promising, as reported by several benchmarking studies. The computational burden, however, limits the application of a full evolutionary model for all sequences and so most algorithms employ heuristics, such as progressive alignment and sequence annealing, while narrowing the scope of the evolutionary model to a pair of sequences. Here, we presented a sequence annealing framework for aligning multiple sequences, which is based on (i) a model for the evolution of three sequences, and (ii) the joint probability of an alignment column as a substitute for the sum-of-pairs approach.

We used a model for the evolution of three sequences, which is represented by a triple-HMM with

seven states that emits three sequences. Biologically more realistic way is to extend the explicit model of sequence evolution, such as the TKF91 [173] or TKF92 [174] models, to describe the evolution of three sequences including the intermediate sequence connecting them (Figure 6.1B). This idea has been successfully applied to the simultaneous estimation of multiple alignment and phylogeny [48]. However, a critical problem of such extension is the huge increase of the number of states in an HMM, which in turn results in high time and memory complexity. Therefore, the method is not appropriate to be used in PEMA, which requires the application of the evolutionary model to every triplet of sequences. Notwithstanding, the evaluation results showed that PEMA achieved good accuracy even with this simple evolutionary model.

The major drawback of PEMA is its high time and memory demands. Given N sequences, PEMA computes the posterior triplewise alignment probabilities by comparing $\binom{N}{3}$ triplets of sequences, and the comparison of each triplet by using an HMM requires $O(N_s^2 L^N)$ time and $O(N_s L^N)$ memory complexity, where N_s is the number of states in the HMM and L is the average length of sequences. However, the complexity was dramatically reduced due to the heuristic method that considers only high contributing triplets of residues to the final likelihood. The other bottleneck is the iterative method for the estimation of the alignment column probability, whose time and memory complexity per iterations are $O(N^3 2^N)$ and $O(2^N)$ respectively. This makes the current version of PEMA only applicable up to around ten sequences in several Kbp length. It currently takes about three minutes for five sequences and one hour for eight sequences in average 1.3Kbp length (on a 2.33 GHz Intel Core2 Duo workstation). FSA and Pecan need less than one minute when aligning eight sequences. This problem, however, can be substantially alleviated by using, for example, (i) the anchor-based alignment technique [14, 18, 124], which first finds an ordered set of local similarities between sequences (anchors) and then aligns the interleaving regions, and (ii) the method of selecting a subset of triplets of sequences to be compared, based on the theory of random graphs [14]. Nevertheless, the importance of this study lies in the fact that this is the first study to use the posterior triplewise alignment probabilities and the estimated column probabilities for aligning multiple sequences, and to show their potential as superior substitutes for current approaches.

6.6 Figures and Tables

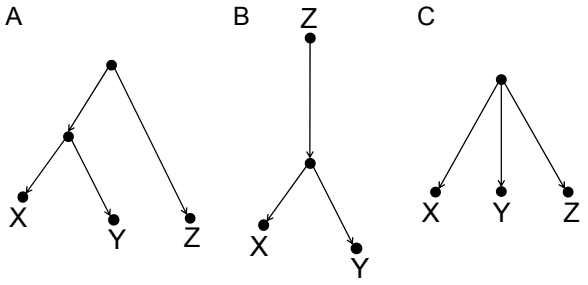


Figure 6.1: Three different phylogenetic trees for the evolution of three sequences X, Y, and Z. (A) A rooted binary tree. (B) An unrooted version based on the reversibility of an underlying Markov process. (C) A simplified tree used in PEMA by assuming a star topology.

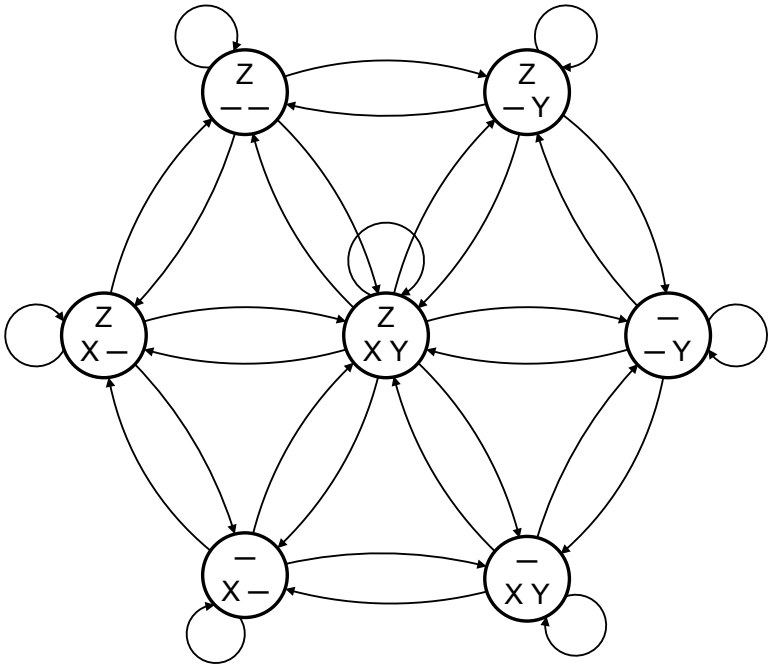


Figure 6.2: Triplet-HMM used by PEMA. Transition probabilities between states are described in Table 6.1. The start and end states are not shown for simplicity.

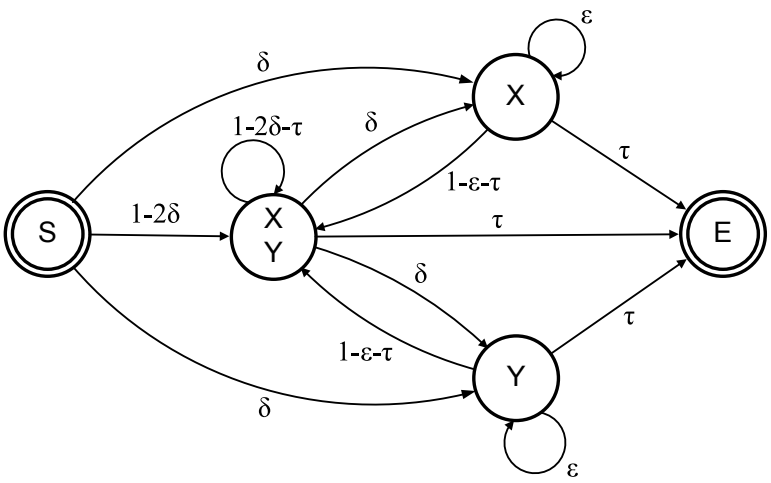


Figure 6.3: Pair-HMM used by PEMA.

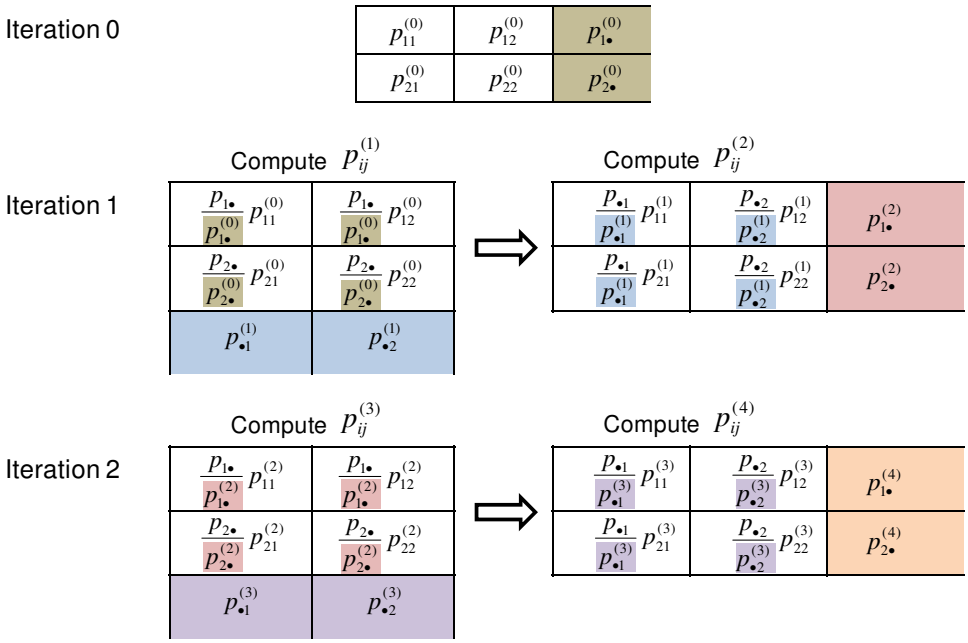


Figure 6.4: Estimation of the cell probabilities of a 2x2 contingency table for which marginal probabilities are given. Here, we want to estimate the cell probabilities p_{ij} by using the given marginal probabilities $p_{i\bullet}$, $p_{\bullet j}$, and the initial cell probabilities $p_{ij}^{(0)} = \pi_{ij}$. At iteration 0, new marginal probabilities $p_{i\bullet}^{(0)}$ for each row are computed from the initial cell probabilities. At iteration 1, $p_{ij}^{(1)}$ are first computed from $p_{ij}^{(0)}$ by the ratio between the given row marginals $p_{i\bullet}$ and the estimated ones $p_{i\bullet}^{(0)}$, and this follows by the computation of the new column marginals $p_{\bullet j}^{(1)}$. Then, $p_{ij}^{(2)}$ are obtained from $p_{ij}^{(1)}$ by the ratio between the given column marginals $p_{\bullet j}$ and the estimated ones $p_{\bullet j}^{(1)}$. The estimated $p_{ij}^{(2)}$ are used to calculate the row marginals $p_{i\bullet}^{(2)}$. Similarly, the cell probabilities $p_{ij}^{(3)}$ and $p_{ij}^{(4)}$ are computed at iteration 2. This process, which incrementally updates the cell probabilities first by the ratio from row marginals and then by the ones from column marginals, repeats until convergence.

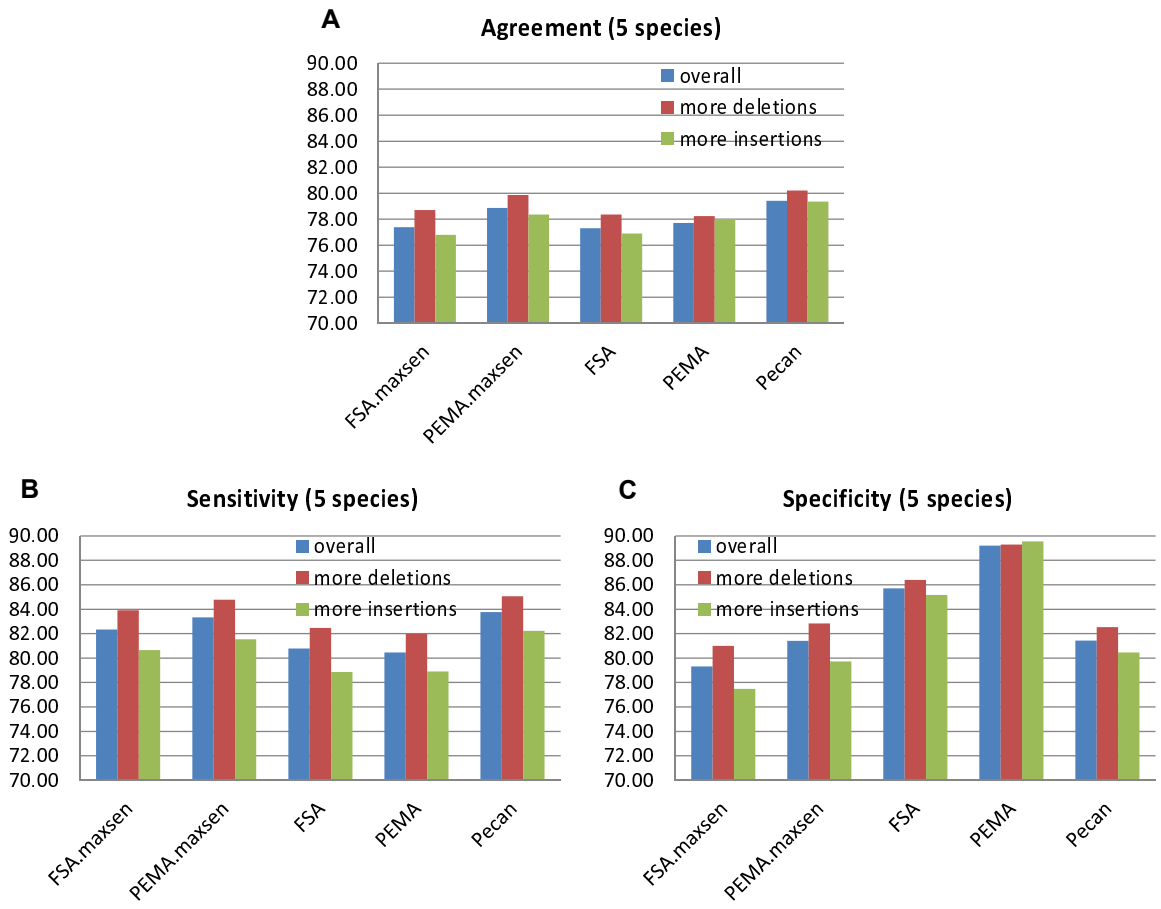


Figure 6.5: Performance of PEMA compared to two existing multiple alignment tools, FSA and Pecan, on five species benchmark. FSA and PEMA were run with two different parameter setting: default and maximum sensitivity ("maxsen") modes. Three evaluation measures, agreement (A), sensitivity (B), and specificity (C) were used to compare the tools. The scores were calculated by using all synthetic data sets (left bars), and by using only data sets where the expected number of insertions is two times more than the number of deletions or vice versa (middle and right bars, respectively).

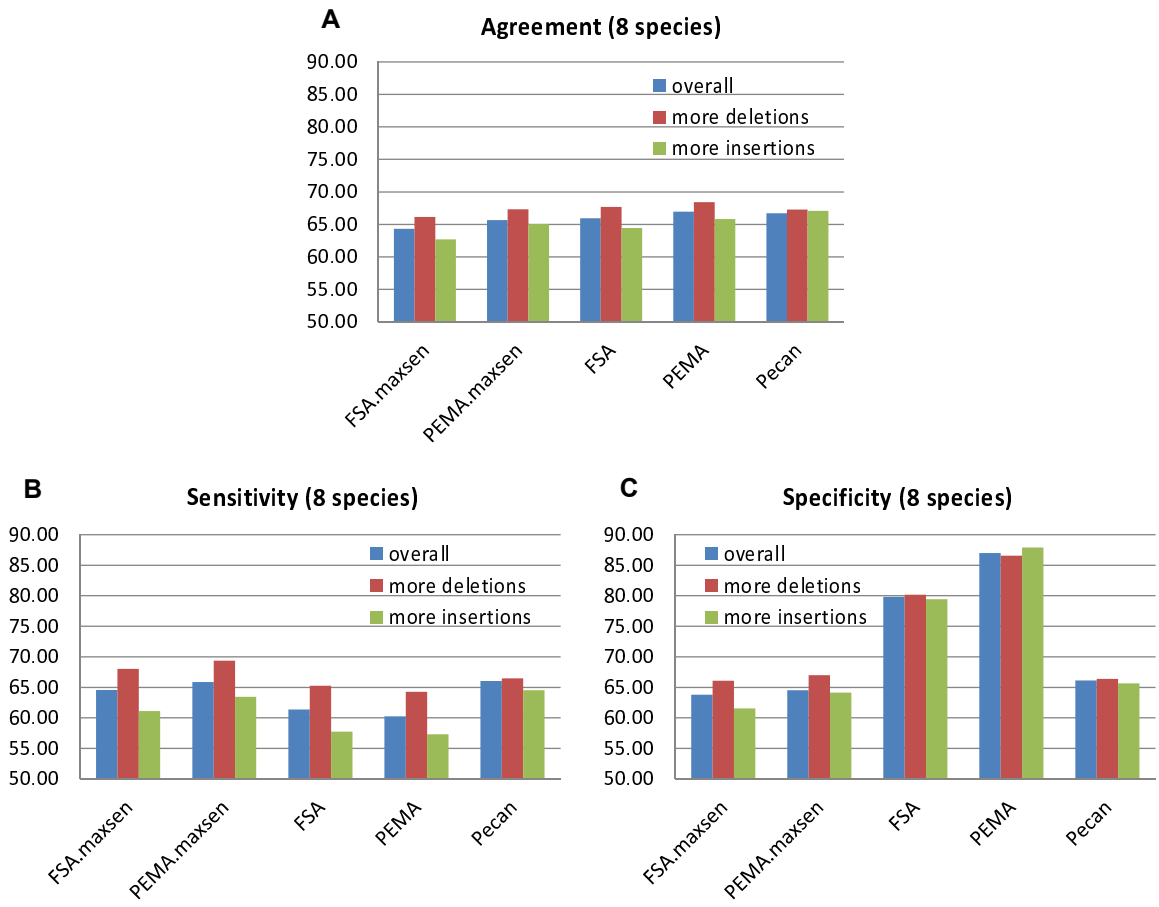


Figure 6.6: Performance of PEMA compared to two existing multiple alignment tools, FSA and Pecan, on eight species benchmark. FSA and PEMA were run with two different parameter setting: default and maximum sensitivity ("maxsen") modes. Three evaluation measures, agreement (A), sensitivity (B), and specificity (C) were used to compare the tools. The scores were calculated by using all synthetic data sets (left bars), and by using only data sets where the expected number of insertions is two times more than the number of deletions or vice versa (middle and right bars, respectively).

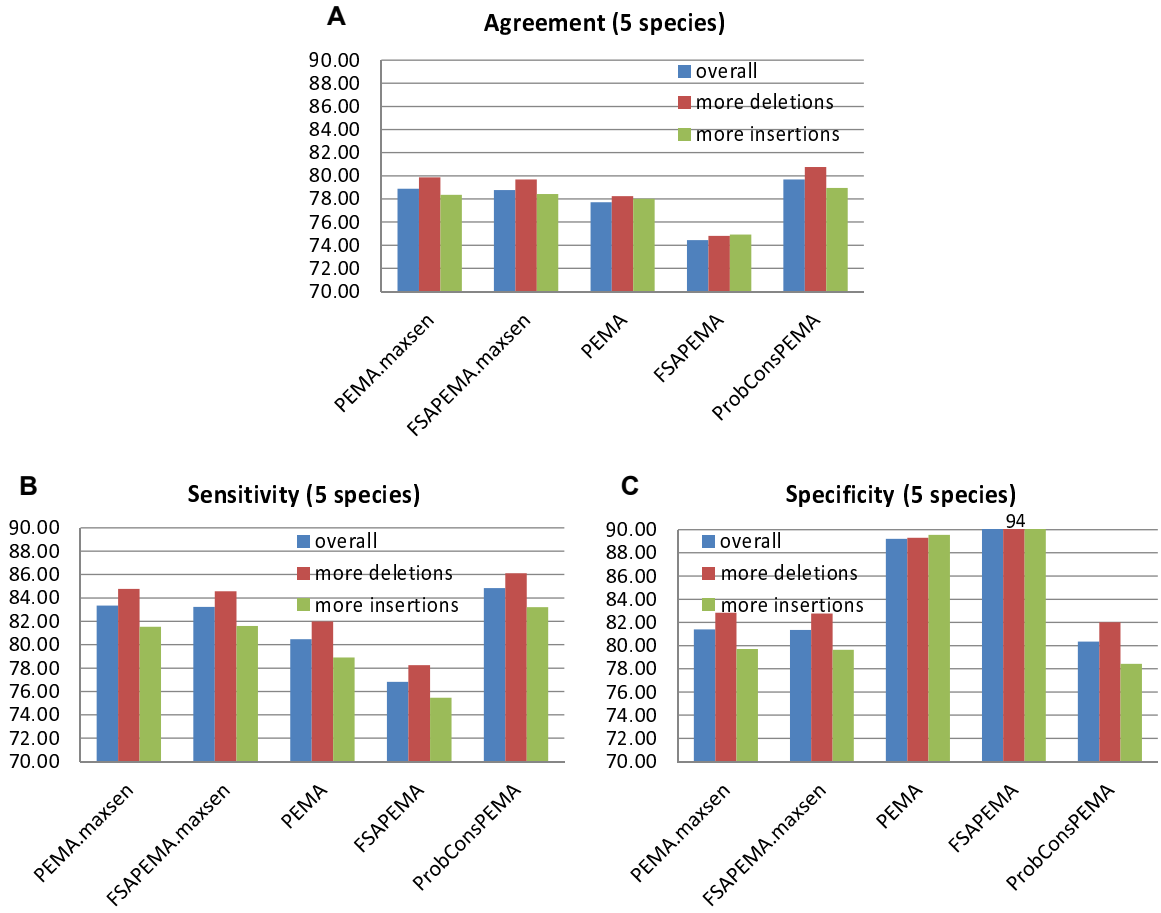


Figure 6.7: Performance of PEMA compared to the variants of other tools on five species benchmark. FSAPEMA and PEMA were run with two different parameter setting: default and maximum sensitivity ("maxsen") modes. Three evaluation measures, agreement (A), sensitivity (B), and specificity (C) were used to compare the tools. The scores were calculated by using all synthetic data sets (left bars), and by using only data sets where the expected number of insertions is two times more than the number of deletions or vice versa (middle and right bars, respectively). The maximum specificity score was set to 90 in the plot and the actual scores of FSAPEMA were 94.20 (overall), 94.22 (more deletions), and 94.21 (more insertions).

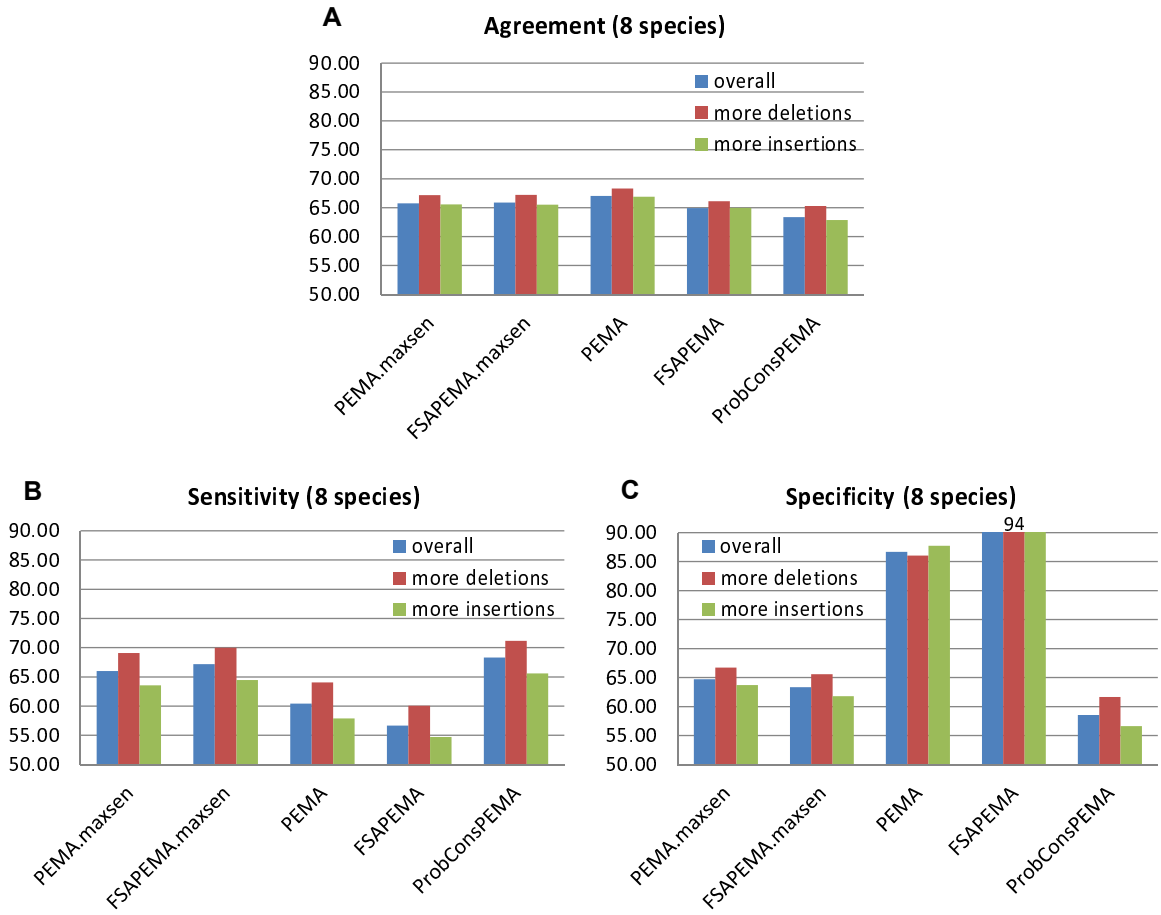


Figure 6.8: Performance of PEMA compared to the variants of other tools on eight species benchmark. FSAPEMA and PEMA were run with two different parameter setting: default and maximum sensitivity ("maxsen") modes. Three evaluation measures, agreement (A), sensitivity (B), and specificity (C) were used to compare the tools. The scores were calculated by using all synthetic data sets (left bars), and by using only data sets where the expected number of insertions is two times more than the number of deletions or vice versa (middle and right bars, respectively). The maximum specificity score was set to 90 in the plot and the actual scores of FSAPEMA were 93.89 (overall), 93.70 (more deletions), and 94.36 (more insertions).

Table 6.1: State transition probabilities of the triple-HMM shown in Figure 6.2.

	S	ZXY	ZX-	Z-Y	Z–	-XY	-X-	–Y	E
S	0	$1-\Sigma^1$	δ_2	δ_2	δ_1	δ_2	δ_1	δ_1	0
ZXY	0	$1-\Sigma^1$	δ_2	δ_2	δ_1	δ_2	δ_1	δ_1	τ
ZX-	0	$1-\Sigma^1$	ε_2	0	σ_1	0	σ_1	0	τ
Z-Y	0	$1-\Sigma^1$	0	ε_2	σ_1	0	0	σ_1	τ
Z–	0	$1-\Sigma^1$	σ_2	σ_2	ε_1	0	0	0	τ
-XY	0	$1-\Sigma^1$	0	0	0	ε_2	σ_1	σ_1	τ
-X-	0	$1-\Sigma^1$	σ_2	0	0	σ_2	ε_1	0	τ
–Y	0	$1-\Sigma^1$	0	σ_2	0	σ_2	0	ε_1	τ

[1]Sum of transition probabilities at the same row except current one

Table 6.2: Evaluation of the iterative method to estimate cell probabilities in a contingency table.

Number of dimensions	4	5	6	7	8
Relative Entropy	0.0025	0.0058	0.0066	0.0056	0.0042

The evaluation process began with the simulation of a contingency table with cell probabilities sampled from the Dirichlet distribution. Then, it computed marginal probabilities for each triplet of dimensions and used the iterative method to re-estimate the cell probabilities, which were next compared to the simulated cell probabilities. The evaluation process was repeated 100 times for each different number of dimensions and the average values of the relative entropy were reported.

Chapter 7

Conclusions

Non-coding regions in genomic DNA are of great significance because (i) they harbor functional elements that are involved in the regulation of gene expression and (ii) they are essential for the study of genome structure and evolution. In this dissertation, we have proposed three novel computational tools for the evolutionary analysis of non-coding sequences and conducted a comparative study of the evolution of regulatory sequences in 12 *Drosophila* species. This dissertation has the following contributions.

- We have presented a probabilistic framework for annotation of insertions and deletions in an alignment of three or more species, and evaluated its performance on synthetic data sets. The framework is also able to realign the given multiple alignment, thereby improving the annotation accuracy significantly. The probabilistic model allows for arbitrary distributions of indel lengths, and a dynamic programming algorithm leads to an efficient implementation that can scale to genome-wide applications.

- We have conducted an empirical evolutionary analysis of a large collection of experimentally verified CRM sequences, taking advantage of the recently sequenced 12 *Drosophila* genomes. Our analysis has revealed several interesting patterns, some along expected lines (but not confirmed previously), and some contrary to our expectations. We believe that our work will furnish evidence orthogonal to experimental characterization for understanding the organizational principles of CRMs, and will be important for developing a theory of regulatory evolution in the future.

- We have developed a novel method for generating benchmarks of non-coding sequence alignments, that relies on a spectrum of parameter values reflecting the genome-wide variation of those parameters. We have shown that our benchmarks can be constructed to mirror

the difficulty of aligning any given set of real genomic sequences. Evaluations on benchmarks constructed to mimic *Drosophila* non-coding sequences suggest a greater accuracy of multiple alignment tools (in this domain) than previously reported, and point to a clear asymmetry in the handling of insertions versus deletions by most alignment tools.

- We have developed a probabilistic framework for aligning multiple sequences that takes advantage of (i) a sequence annealing algorithm, (ii) a model for the evolution of three sequences, and (iii) the joint probability of an alignment column as a substitute for the traditionally used sum-of-pairs score. The evaluation results demonstrate that the new framework produces alignments of much greater specificity than state-of-the-art methods, without compromising too much in terms of sensitivity.

Extant sequences are the results of evolutionary processes of nucleotide substitution, insertion and deletion. As a result, the task of tracing the history of evolutionary events naturally leads to the annotation of insertions and deletions, as well as the alignment of sequences. This suggests, as a future direction, the integration of the tasks of aligning sequences and annotating insertion and deletion events. This could be achieved by using an explicit model of sequence evolution that can distinguish insertions from deletions, such as the TKF91 [173] and TKF92 [174] models, with heuristics that can reduce computational burden of such probabilistic models, such as techniques to reduce recursions in hidden Markov models [101, 159] and the Markov chain Monte Carlo (MCMC) approach [66].

As more and more genome sequences become available, bioinformatics tools are increasingly important for the comparison and analysis of non-coding sequences across different species. They will play a significant role in solving many biological problems and further contribute to broaden our understanding of organismal diversity and evolution.

References

[1] S. E. Ahnert, T. M. Fink, and A. Zinovyev. How much non-coding dna do eukaryotes require? *J Theor Biol*, 252(4):587–92, Jun 2008.

[2] P. Andolfatto. Adaptive evolution of non-coding dna in drosophila. *Nature*, 437(7062):1149–52, Oct 2005.

[3] D. N. Arnosti. Analysis and function of transcriptional regulatory elements: insights from drosophila. *Annu Rev Entomol*, 48:579–602, 2003.

[4] S. Batzoglou. The many faces of sequence alignment. *Brief Bioinform*, 6(1):6–22, Mar 2005.

[5] G. Bejerano, M. Pheasant, I. Makunin, S. Stephen, W. J. Kent, J. S. Mattick, et al. Ultraconserved elements in the human genome. *Science*, 304(5675):1321–5, May 2004.

[6] C. M. Bergman and M. Kreitman. Analysis of conserved noncoding dna in drosophila reveals similar constraints in intergenic and intronic sequences. *Genome Res*, 11(8):1335–45, Aug 2001.

[7] C. M. Bergman, J. W. Carlson, and S. E. Celniker. Drosophila dnase i footprint database: a systematic genome annotation of transcription factor binding sites in the fruitfly, drosophila melanogaster. *Bioinformatics*, 21(8):1747–9, Apr 2005.

[8] M. J. Bishop and E. A. Thompson. Maximum likelihood alignment of dna sequences. *J Mol Biol*, 190(2):159–65, July 1986.

[9] E. M. Blackwood and J. T. Kadonaga. Going the distance: a current view of enhancer action. *Science*, 281(5373):60–3, Jul 1998.

[10] M. Blanchette, E. D. Green, W. Miller, and D. Haussler. Reconstructing large regions of an ancestral mammalian genome in silico. *Genome Res*, 14(12):2412–23, Dec 2004.

[11] M. Blanchette, W. J. Kent, C. Riemer, L. Elnitski, A. F. Smit, K. M. Roskin, et al. Aligning multiple genomic sequences with the threaded blockset aligner. *Genome Res*, 14(4):708–15, Apr 2004.

[12] A. R. Borneman, T. A. Gianoulis, Z. D. Zhang, H. Yu, J. Rozowsky, M. R. Seringhaus, et al. Divergence of transcription factor binding sites across related yeast species. *Science*, 317 (5839):815–9, Aug 2007.

[13] R. K. Bradley and I. Holmes. Transducers: an emerging probabilistic framework for modeling indels on trees. *Bioinformatics*, 23(23):3258–62, Dec 2007.

[14] R. K. Bradley, A. Roberts, M. Smoot, S. Juvekar, J. Do, C. Dewey, et al. Fast statistical alignment. *PLoS Comput Biol*, 5(5):e1000392, May 2009.

[15] N. Bray and L. Pachter. Mavid: constrained ancestral alignment of multiple sequences. *Genome Res*, 14(4):693–9, Apr 2004.

[16] C. D. Brown, D. S. Johnson, and A. Sidow. Functional architecture and evolution of transcriptional elements that drive gene coexpression. *Science*, 317(5844):1557–60, Sep 2007.

[17] C. T. Brown and Jr. Callan C. G. Evolutionary comparisons suggest many novel camp response protein binding sites in escherichia coli. *Proc Natl Acad Sci U S A*, 101(8):2404–9, Feb 2004.

[18] M. Brudno, C. B. Do, G. M. Cooper, M. F. Kim, E. Davydov, E. D. Green, et al. Analysis of conserved noncoding dna in drosophila reveals similar constraints in intergenic and intronic sequences. *Genome Res.*, 13:721–731, 2003.

[19] R. A. Cameron, S. H. Chow, K. Berney, T. Chiu, Q. Yuan, A. Krämer, et al. An evolutionary constraint: strongly disfavored class of change in dna sequence during divergence of cis-regulatory modules. *Proc Natl Acad Sci U S A*, 102(33):11769–74, Aug 2005.

[20] S. Carroll, J. Grenier, and S. Weatherbee. *From DNA to Diversity: Molecular Genetics and the Evolution of Animal Design*. Blackwell Science, 2001.

[21] R. A. Cartwright. Dna assembly with gaps (dawg): simulating sequence evolution. *Bioinformatics*, 21 Suppl 3:iii31–8, Nov 2005.

[22] F. C. Chen, C. J. Chen, and T. J. Chuang. Indelscan: a web server for comparative identification of species-specific and non-species-specific insertion/deletion events. *Nucleic Acids Res*, 35(Web Server issue):W633–8, Jul 2007.

[23] L. Chindelevitch, Z. Li, E. Blais, and M. Blanchette. On the inference of parsimonious indel evolutionary scenarios. *J Bioinform Comput Biol*, 4(3):721–44, Jun 2006.

[24] A. G. Clark, M. B. Eisen, D. R. Smith, C. M. Bergman, B. Oliver, T. A. Markow, et al. Evolution of genes and genomes on the drosophila phylogeny. *Nature*, 450(7167):203–18, Nov 2007.

[25] T. G. Clark, T. Andrew, G. M. Cooper, E. H. Margulies, J. C. Mullikin, and D. J. Balding. Functional constraint and small insertions and deletions in the encode regions of the human genome. *Genome Biol*, 8(9):R180, 2007.

[26] D. E. Clyde, M. S. G. Corado, X. Wu, A. Paré, D. Papatsenko, and S. Small. A self-organizing system of repressor gradients establishes segmental complexity in drosophila. *Nature*, 426 (6968):849–53, Dec 2003.

[27] J. M. Comeron and M. Kreitman. The correlation between intron length and recombination in drosophila. dynamic equilibrium between mutational and selective forces. *Genetics*, 156 (3):1175–90, Nov 2000.

[28] J. Costas, F. Casares, and J. Vieira. Turnover of binding sites for transcription factors involved in early drosophila development. *Gene*, 310:215–20, May 2003.

[29] D. Das, N. Banerjee, and M. Q. Zhang. Interacting models of cooperative gene regulation. *Proc Natl Acad Sci U S A*, 101(46):16234–9, Nov 2004.

[30] E. H. Davidson. *The Regulatory Genome: Gene Regulatory Networks in Development and Evolution*. Academic Press, 2006.

[31] C. R. Dearolf, J. Topol, and C. S. Parker. The caudal gene product is a direct activator of fushi tarazu transcription during drosophila embryogenesis. *Nature*, 341(6240):340–3, Sep 1989.

[32] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via em algorithm. *Journal of the Royal Statistical Society Series B (Methodological)*, 39:1–38, 1977.

[33] E. T. Dermitzakis, C. M. Bergman, and A. G. Clark. Tracing the evolutionary history of drosophila regulatory regions with models that identify transcription factor binding sites. *Mol Biol Evol*, 20(5):703–14, May 2003.

[34] C. N. Dewey and L. Pachter. Evolution at the nucleotide level: the problem of multiple whole-genome alignment. *Hum Mol Genet*, 15 Spec No 1:R51–6, Apr 2006.

[35] A. B. Diallo, V. Makarenkov, and M. Blanchette. Exact and heuristic algorithms for the indel maximum likelihood problem. *J Comput Biol*, 14(4):446–61, May 2007.

[36] J. L. Diaz, T. Oltersdorf, W. Horne, M. McConnell, G. Wilson, S. Weeks, et al. A common binding site mediates heterodimerization and homodimerization of bcl-2 family members. *J Biol Chem*, 272(17):11350–11355, Apr 1997.

[37] C. B. Do, M. S. P. Mahabhashyam, M. Brudno, and S. Batzoglou. Probcons: Probabilistic consistency-based multiple sequence alignment. *Genome Res*, 15(2):330–40, Feb 2005.

[38] S. W. Doniger and J. C. Fay. Frequent gain and loss of functional transcription factor binding sites. *PLoS Comput Biol*, 3(5):e99, May 2007.

[39] A. W. M. Dress, C. Flamm, G. Fritzsch, S. Grünewald, M. Kruspe, S. J. Prohaska, et al. Noisy: identification of problematic columns in multiple sequence alignments. *Algorithms Mol Biol*, 3:7, 2008.

[40] Drosophila 12 Genomes Consortium. Evolution of genes and genomes on the drosophila phylogeny. *Nature*, 450(7167):203–18, Nov 2007.

[41] R. C. Edgar and S. Batzoglou. Multiple sequence alignment. *Curr Opin Struct Biol*, 16(3): 368–73, Jun 2006.

[42] L. Elnitski, V. X. Jin, P. J. Farnham, and S. J. M. Jones. Locating mammalian transcription factor binding sites: a survey of computational and experimental techniques. *Genome Res*, 16(12):1455–64, Dec 2006.

[43] E. Emberly, N. Rajewsky, and E. D. Siggia. Conservation of regulatory elements between two species of drosophila. *BMC Bioinformatics*, 4:57, Nov 2003.

[44] J. Felsenstein. Evolutionary trees from dna sequences: a maximum likelihood approach. *J Mol Evol*, 17(6):368–76, 1981.

[45] J. Felsenstein and G. A. Churchill. A hidden markov model approach to variation among sites in rate of evolution. *Mol Biol Evol*, 13(1):93–104, Jan 1996.

[46] D. F. Feng and R. F. Doolittle. Progressive sequence alignment as a prerequisite to correct phylogenetic trees. *J Mol Evol*, 25(4):351–60, 1987.

[47] S. E. Fienberg. An iterative procedure for estimation in contingency tables. *The Annals of Mathematical Statistics*, 41(3):907–917, 1970.

[48] R. Fleissner, D. Metzler, and A. von Haeseler. Simultaneous statistical multiple alignment and phylogeny reconstruction. *Syst Biol*, 54(4):548–61, Aug 2005.

[49] W. Fletcher and Z. Yang. Indelible: a flexible simulator of biological sequence evolution. *Mol Biol Evol*, 26(8):1879–88, Aug 2009.

[50] J. Fredslund, J. Hein, and T. Scharling. A large version of the small parsimony problem. In Gary Benson and Roderic D. M. Page, editors, *Proceedings of the 4th Workshop on Algorithms in Bioinformatics (WABI)*, volume 2812 of *Lecture Notes in Computer Science*, pages 417–432. Springer, 2004.

[51] I. J. Good. Maximum entropy for hypothesis formulation, especially for multidimensional contingency tables. *The Annals of Mathematical Statistics*, 34(3):911–934, 1963.

[52] S. Gray and M. Levine. Transcriptional repression in development. *Curr Opin Cell Biol*, 8 (3):358–64, Jun 1996.

[53] M. S. Halfon, S. M. Gallo, and C. M. Bergman. Redfly 2.0: an integrated database of cis-regulatory modules and transcription factor binding sites in drosophila. *Nucleic Acids Res*, 36(Database issue):D594–8, Jan 2008.

[54] B. G. Hall. How well does the hot score reflect sequence alignment accuracy? *Mol Biol Evol*, 25(8):1576–80, Aug 2008.

[55] D. L. Halligan, A. Eyre-Walker, P. Andolfatto, and P. D. Keightley. Patterns of evolutionary constraints in intronic and intergenic dna of drosophila. *Genome Res*, 14(2):273–9, Feb 2004.

[56] A. L. Halpern and W. J. Bruno. Evolutionary distances for protein-coding sequences: modeling site-specific residue frequencies. *Mol Biol Evol*, 15(7):910–7, Jul 1998.

[57] R. C. Hardison. Comparative genomics. *PLoS Biol*, 1(2):E58, Nov 2003.

[58] E. E. Hare, B. K. Peterson, V. N. Iyer, R. Meier, and M. B. Eisen. Sepsid even-skipped enhancers are functionally conserved in drosophila despite lack of sequence conservation. *PLoS Genet*, 4(6):e1000106, Jun 2008.

[59] M. Hasegawa, H. Kishino, and T. Yano. Dating of the human-ape splitting by a molecular clock of mitochondrial dna. *J Mol Evol*, 22(2):160–74, 1985.

[60] X. He, X. Ling, and S. Sinha. Alignment and prediction of cis-regulatory modules based on a probabilistic model of evolution. *PLoS Comput Biol*, 5(3):e1000299, Mar 2009.

[61] J. Hein. An algorithm for statistical alignment of sequences related by a binary tree. *Pac Symp Biocomput*, pages 179–90, 2001.

[62] J. Hein, C. Wiuf, B. Knudsen, M. B. Moller, and G. Wibling. Statistical alignment: computational properties, homology testing and goodness-of-fit. *J Mol Biol*, 302(1):265–79, Sep 2000.

[63] J. Hein, J. L. Jensen, and C. N. Pedersen. Recursions for statistical multiple alignment. *Proc Natl Acad Sci U S A*, 100(25):14960–5, Dec 2003.

[64] M. M. Hoffman and E. Birney. Estimating the neutral rate of nucleotide substitution using introns. *Mol Biol Evol*, 24(2):522–31, Feb 2007.

[65] I. Holmes. Using guide trees to construct multiple-sequence evolutionary hmms. *Bioinformatics*, 19 Suppl 1:i147–57, 2003.

[66] I. Holmes and W.J. Bruno. Evolutionary hmms: a bayesian approach to multiple alignment. *Bioinformatics*, 17(9):803–820, Sep 2001.

[67] M. L. Howard and E. H. Davidson. cis-regulatory control circuits in development. *Dev Biol*, 271(1):109–18, Jul 2004.

[68] W. Huang, J. R. Nevins, and U. Ohler. Phylogenetic simulation of promoter evolution: estimation and modeling of binding site turnover events and assessment of their impact on alignment tools. *Genome Biol*, 8(10):R225, 2007.

[69] C. T. Ireland and S. Kullback. Contingency tables with given marginals. *Biometrika*, 55(1): 179–188, 1968.

[70] C. T. Ireland and S. Kullback. Minimum discrimination information estimation. *Biometrics*, 24(3):707–13, Sep 1968.

[71] H. Janssens, S. Hou, J. Jaeger, A. Kim, E. Myasnikova, D. Sharp, et al. Quantitative and predictive model of transcriptional control of the drosophila melanogaster even skipped gene. *Nat Genet*, 38(10):1159–65, Oct 2006.

[72] E. T. Jaynes. Information theory and statistical mechanics. *Physical Review*, 106:620, 1957.

[73] T. H. Jukes and C. R. Cantor. *Evolution of protein molecules, In Mammalian protein metabolism (ed. H.N. Munro).* Academic Press, 1969.

[74] D. Karolchik, R. M. Kuhn, R. Baertsch, G. P. Barber, H. Clawson, M. Diekhans, et al. The ucsc genome browser database: 2008 update. *Nucleic Acids Res*, 36(Database issue):D773–9, Jan 2008.

[75] K. Katoh and H. Toh. Recent developments in the mafft multiple sequence alignment program. *Brief Bioinform*, 9(4):286–98, Jul 2008.

[76] K. Katoh, K. Kuma, H. Toh, and T. Miyata. Mafft version 5: improvement in accuracy of multiple sequence alignment. *Nucleic Acids Res.*, 33(2):511–518, 2005.

[77] P. D. Keightley and T. Johnson. Mcalign: Stochastic alignment of noncoding dna sequences based on an evolutionary model of sequence evolution. *Genome Res.*, 14(3):442–450, Mar 2004.

[78] C. Kemena and C. Notredame. Upcoming challenges for multiple sequence alignment methods in the high-throughput era. *Bioinformatics*, 25(19):2455–65, Oct 2009.

[79] J. Kim. Macro-evolution of the hairy enhancer in drosophila species. *J Exp Zool*, 291(2): 175–85, Aug 2001.

[80] J. Kim and S. Sinha. Indelign: a probabilistic framework for annotation of insertions and deletions in a multiple alignment. *Bioinformatics*, 23(3):289–97, Feb 2007.

[81] J. Kim, X. He, and S. Sinha. Evolution of regulatory sequences in 12 drosophila species. *PLoS Genet*, 5(1):e1000330, Jan 2009.

[82] M. Kimura. A simple method for estimating evolutionary rates of base substitutions through comparative studies of nucleotide sequences. *J. Mol. Evol.*, 16(2):111–120, Dec 1980.

[83] M. Kimura. *The neutral theory of molecular evolution*. Cambridge University Press, 1983.

[84] M. M. Kulkarni and D. N. Arnosti. cis-regulatory logic of short-range transcriptional repression in drosophila melanogaster. *Mol Cell Biol*, 25(9):3411–3420, May 2005.

[85] S. Kumar and A. Filipski. Multiple sequence alignment: in pursuit of homologous dna positions. *Genome Res*, 17(2):127–35, Feb 2007.

[86] G. Landan and D. Graur. Heads or tails: a simple reliability check for multiple sequence alignments. *Mol Biol Evol*, 24(6):1380–3, Jun 2007.

[87] G. Landan and D. Graur. Local reliability measures from sets of co-optimal multiple sequence alignments. *Pac Symp Biocomput*, pages 15–24, 2008.

[88] M. A. Larkin, G. Blackshields, N. P. Brown, R. Chenna, P. A. McGettigan, H. McWilliam, et al. Clustal w and clustal x version 2.0. *Bioinformatics*, 23(21):2947–8, Nov 2007.

[89] D. Lebrecht, M. Foehr, E. Smith, F. J. P. Lopes, C. E. Vanario-Alonso, J. Reinitz, et al. Bicoid cooperative dna binding is critical for embryonic patterning in drosophila. *Proc Natl Acad Sci U S A*, 102(37):13176–81, Sep 2005.

[90] L. Li, Q. Zhu, X. He, S. Sinha, and M. S. Halfon. Large-scale analysis of transcriptional cis-regulatory modules reveals both common features and distinct subclasses. *Genome Biol*, 8(6):R101, 2007.

[91] X. Y. Li, S. MacArthur, R. Bourgon, D. Nix, D. A. Pollard, V. N. Iyer, et al. Transcription factors bind thousands of active and inactive regions in the drosophila blastoderm. *PLoS Biol*, 6(2):e27, Feb 2008.

[92] A. P. Lifanov, V. J. Makeev, A. G. Nazina, and D. A. Papatsenko. Homotypic regulatory clusters in drosophila. *Genome Res*, 13(4):579–88, Apr 2003.

[93] G. G. Loots and I. Ovcharenko. rvista 2.0: evolutionary analysis of transcription factor binding sites. *Nucleic Acids Res*, 32(Web Server issue):W217–21, Jul 2004.

[94] A. Löytynoja and N. Goldman. An algorithm for progressive multiple alignment of sequences with insertions. *Proc Natl Acad Sci U S A*, 102(30):10557–62, Jul 2005.

[95] A. Löytynoja and N. Goldman. Phylogeny-aware gap placement prevents errors in sequence alignment and evolutionary analysis. *Science*, 320(5883):1632–5, Jun 2008.

[96] M. Z. Ludwig. Functional evolution of noncoding dna. *Curr Opin Genet Dev*, 12(6):634–9, Dec 2002.

[97] M. Z. Ludwig, N. H. Patel, and M. Kreitman. Functional analysis of eve stripe 2 enhancer evolution in drosophila: rules governing conservation and change. *Development*, 125(5):949–58, Mar 1998.

[98] M. Z. Ludwig, A. Palsson, E. Alekseeva, C. M. Bergman, J. Nathan, and M. Kreitman. Functional evolution of a cis-regulatory module. *PLoS Biol*, 3(4):e93, Apr 2005.

[99] G. Lunter, C. P. Ponting, and J. Hein. Genome-wide identification of human functional dna using a neutral indel model. *PLoS Comput Biol*, 2(1):e5, Jan 2006.

[100] G. Lunter, A. Rocco, N. Mimouni, A. Heger, A. Caldeira, and J. Hein. Uncertainty in homology inferences: assessing and improving genomic sequence alignment. *Genome Res*, 18 (2):298–309, Feb 2008.

[101] G. A. Lunter, I. Miklós, Y. S. Song, and J. Hein. An efficient algorithm for statistical multiple alignment on arbitrary phylogenetic trees. *J Comput Biol*, 10(6):869–89, 2003.

[102] G. A. Lunter, A. J. Drummond, I. Miklós, and J. Hein. *Statistical Alignment: Recent Progress, New Applications, and Challenges, in: Rasmus Nielsen (ed.), Statistical methods in Molecular Evolution*. Springer Verlag, 2004.

[103] G. A. Lunter, I. Miklós, and I. Holmes. National human genome research institute. 2004. Fruitfly Genome Sequencing.

[104] S. Mahony, D. L. Corcoran, E. Feingold, and P. V. Benos. Regulatory conservation of protein coding and microrna genes in vertebrates: lessons from the opossum genome. *Genome Biol*, 8(5):R84, 2007.

[105] L. Martignetti, M. Caselle, B. Jacq, and C. Herrmann. Drosocb: a high resolution map of conserved non coding sequences in drosophila. Technical Report DFTT 4/07, Oct 2007.

[106] D. Metzler. Statistical alignment based on fragment insertion and deletion models. *Bioinformatics*, 19(4):490–9, Mar 2003.

[107] I. Miklós, G. A. Lunter, and I. Holmes. A "long indel" model for evolutionary sequence alignment. *Mol Biol Evol*, 21(3):529–40, Mar 2004.

[108] W. Miller, K. D. Makova, A. Nekrutenko, and R. C. Hardison. Comparative genomics. *Annu Rev Genomics Hum Genet*, 5:15–56, 2004.

[109] K. Mizuguchi, C. M. Deane, T. L. Blundell, and J. P. Overington. Homstrad: a database of protein structure alignments for homologous families. *Protein Sci*, 7(11):2469–71, Nov 1998.

[110] B. Morgenstern. Dialign: multiple dna and protein sequence alignment at bibiserv. *Nucleic Acids Res*, 32(Web Server issue):W33–6, Jul 2004.

[111] B. Morgenstern. Dialign 2: improvement of the segment-to-segment approach to multiple sequence alignment. *Bioinformatics*, 15(3):211–8, Mar 1999.

[112] A. M. Moses, D. Y. Chiang, M. Kellis, E. S. Lander, and M. B. Eisen. Position specific variation in the rate of evolution in transcription factor binding sites. *BMC Evol Biol*, 3:19, Aug 2003.

[113] A. M. Moses, D. Y. Chiang, D. A. Pollard, V. N. Iyer, and M. B. Eisen. Monkey: identifying conserved transcription-factor binding sites in multiple alignments using a binding site-specific evolutionary model. *Genome Biol*, 5(12):R98, 2004.

[114] A. M. Moses, D. A. Pollard, D. A. Nix, V. N. Iyer, X. Li, M. D. Biggin, et al. Large-scale turnover of functional transcription factor binding sites in drosophila. *PLoS Comput Biol*, 2 (10):e130, Oct 2006.

[115] V. Mustonen and M. Lässig. Evolutionary population genetics of promoters: predicting binding sites and functional phylogenies. *Proc Natl Acad Sci U S A*, 102(44):15936–41, Nov 2005.

[116] S. B. Needleman and C. D. Wunsch. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J Mol Biol*, 48(3):443–53, Mar 1970.

[117] C. Notredame. Recent progress in multiple sequence alignment: a survey. *Pharmacogenomics*, 3(1):131–44, January 2002.

[118] C. Notredame. Recent evolutions of multiple sequence alignment algorithms. *PLoS Comput Biol*, 3(8):e123, Aug 2007.

[119] T. H. Ogdenw and M. S. Rosenberg. Multiple sequence alignment accuracy and phylogenetic inference. *Syst Biol*, 55(2):314–28, Apr 2006.

[120] D. Papatsenko and M. S. Levine. Dual regulation by the hunchback gradient in the drosophila embryo. *Proc Natl Acad Sci U S A*, 105(8):2901–2906, Feb 2008.

[121] J. Parsch. Selective constraints on intron evolution in drosophila. *Genetics*, 165(4):1843–51, Dec 2003.

[122] B. Paten, J. Herrero, K. Beal, S. Fitzgerald, and E. Birney. Enredo and pecan: Genome-wide mammalian consistency-based multiple alignment with paralogs. *Genome Res*, Oct 2008.

[123] B. Paten, J. Herrero, S. Fitzgerald, K. Beal, P. Flicek, I. Holmes, et al. Genome-wide nucleotide-level mammalian ancestor reconstruction. *Genome Res*, 18(11):1829–43, Nov 2008.

[124] B. Paten, J. Herrero, K. Beal, and E. Birney. Sequence progressive alignment, a framework for practical large-scale probabilistic consistency alignment. *Bioinformatics*, 25(3):295–301, Feb 2009.

[125] D. A. Petrov. Dna loss and evolution of genome size in drosophila. *Genetica*, 115(1):81–91, May 2002.

[126] D. A. Petrov and D. L. Hartl. High rate of dna loss in the drosophila melanogaster and drosophila virilis species groups. *Mol Biol Evol*, 15(3):293–302, Mar 1998.

[127] D. A. Petrov and D. L. Hartl. Pseudogene evolution and natural selection for a compact genome. *J Hered*, 91(3):221–7, May-Jun 2000.

[128] D. A. Petrov, E. R. Lozovskaya, and D. L. Hartl. High intrinsic rate of dna loss in drosophila. *Nature*, 384(6607):346–9, Nov 1996.

[129] D. A. Petrov, T. A. Sangster, J. S. Johnston, D. L. Hartl, and K. L. Shaw. Evidence for dna loss as a determinant of genome size. *Science*, 287(5455):1060–2, Feb 2000.

[130] A. Phillips, D. Janies, and W. Wheeler. Multiple sequence alignment in phylogenetic analysis. *Mol Phylogenet Evol*, 16(3):317–30, September 2000.

[131] W. Pirovano and J. Heringa. Multiple sequence alignment. *Methods Mol Biol*, 452:143–61, 2008.

[132] D. A. Pollard, C. M. Bergman, J. Stoye, S. E. Celniker, and M. B. Eisen. Benchmarking tools for the alignment of functional noncoding dna. *BMC Bioinformatics*, 5:6, Jan 2004.

[133] D. A. Pollard, A. M. Moses, V. N. Iyer, and M. B. Eisen. Detecting the limits of regulatory element conservation and divergence estimation using pairwise and multiple alignments. *BMC Bioinformatics*, 7:376, 2006.

[134] A. Prakash and M. Tompa. Statistics of local multiple alignments. *Bioinformatics*, 21 Suppl 1:i344–50, Jun 2005.

[135] A. Prakash and M. Tompa. Measuring the accuracy of genome-size multiple alignments. *Genome Biol*, 8(6):R124, 2007.

[136] D. C. Presgraves. Intron length evolution in drosophila. *Mol Biol Evol*, 23(11):2203–13, Nov 2006.

[137] S. E. Ptak and D. A. Petrov. How intron splicing affects the deletion and insertion profile in drosophila melanogaster. *Genetics*, 162(3):1233–44, Nov 2002.

[138] L. R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. In *Proceedings of the IEEE*, volume 77, pages 257–286, 1989.

[139] A. Rambaut and N. C. Grassly. Seq-gen: an application for the monte carlo simulation of dna sequence evolution along phylogenetic trees. *Comput Appl Biosci*, 13(3):235–8, Jun 1997.

[140] E. Rivas. Evolutionary models for insertions and deletions in a probabilistic modeling framework. *BMC Bioinformatics*, 6:63, 2005.

[141] E. Rivas and S. R. Eddy. Probabilistic phylogenetic inference with insertions and deletions. *PLoS Comput Biol*, 4(9):e1000172, 2008.

[142] R. Rivera-Pomar, X. Lu, N. Perrimon, H. Taubert, and H. Jäckle. Activation of posterior gap gene expression in the drosophila blastoderm. *Nature*, 376(6537):253–6, Jul 1995.

[143] A. Rokas. Genomics. lining up to avoid bias. *Science*, 319(5862):416–7, Jan 2008.

[144] M. S. Rosenberg. Multiple sequence alignment accuracy and evolutionary distance estimation. *BMC Bioinformatics*, 6:278, 2005.

[145] U. Roshan and D. R. Livesay. Probalign: multiple sequence alignment using partition function posterior probabilities. *Bioinformatics*, 22(22):2715–21, November 2006.

[146] J. Ryu, K. Nam, C. Oh, H. Nam, S. Kim, J. Yoon, et al. The homeobox gene caudal regulates constitutive local expression of antimicrobial peptide genes in drosophila epithelia. *Mol Cell Biol*, 24(1):172–85, Jan 2004.

[147] M. D. Schroeder, M. Pearce, J. Fak, H. Fan, U. Unnerstall, E. Emberly, et al. Transcriptional control in the segmentation gene network of drosophila. *PLoS Biol*, 2(9):E271, Sep 2004.

[148] A. S. Schwartz and L. Pachter. Multiple alignment by sequence annealing. *Bioinformatics*, 23(2):e24–9, Jan 2007.

[149] A. S. Schwartz, E. W. Myers, and L. Pachter. Alignment metric accuracy, October 2005 arXiv: q-bio.QM/0510052.

[150] S. Schwartz, W. J. Kent, A. Smit, Z. Zhang, R. Baertsch, R. C. Hardison, et al. Human-mouse alignments with blastz. *Genome Res*, 13(1):103–7, Jan 2003.

[151] A. Siepel, G. Bejerano, J. S. Pedersen, A. S. Hinrichs, M. Hou, K. Rosenbloom, et al. Evolutionarily conserved elements in vertebrate, insect, worm, and yeast genomes. *Genome Res*, 15(8):1034–50, Aug 2005.

[152] V. Simossis, J. Kleinjung, and J. Heringa. An overview of multiple sequence alignment. *Curr Protoc Bioinformatics*, Chapter 3:Unit 3 7, Nov 2003.

[153] S. Sinha and X. He. Morph: probabilistic alignment combined with hidden markov models of cis-regulatory modules. *PLoS Comput Biol*, 3(11):e216, Nov 2007.

[154] S. Sinha and E. D. Siggia. Sequence turnover and tandem repeats in cis-regulatory modules in drosophila. *Mol Biol Evol*, 22(4):874–85, Apr 2005.

[155] S. Sinha, E. van Nimwegen, and E. D. Siggia. A probabilistic method to detect regulatory modules. *Bioinformatics*, 19 Suppl 1:i292–301, 2003.

[156] S. Sinha, M. D. Schroeder, U. Unnerstall, U. Gaul, and E. D. Siggia. Cross-species comparison significantly improves genome-wide prediction of cis-regulatory modules in drosophila. *BMC Bioinformatics*, 5:129, Sep 2004.

[157] S. Small, A. Blair, and M. Levine. Regulation of even-skipped stripe 2 in the drosophila embryo. *EMBO J*, 11(11):4047–57, Nov 1992.

[158] S. Snir and L. Pachter. Phylogenetic profiling of insertions and deletions in vertebrate genomes. In Alberto Apostolico, Concettina Guerra, Sorin Istrail, Pavel A. Pevzner, and Michael S. Waterman, editors, *RECOMB*, volume 3909 of *Lecture Notes in Computer Science*, pages 265–280. Springer, 2006.

[159] Y. S. Song. A sufficient condition for reducing recursions in hidden markov models. *Bull Math Biol*, 68(2):361–84, February 2006.

[160] A. Stark, M. F. Lin, P. Kheradpour, J. S. Pedersen, L. Parts, J. W. Carlson, et al. Discovery of functional elements in 12 drosophila genomes using evolutionary signatures. *Nature*, 450 (7167):219–32, Nov 2007.

[161] M. Steel and J. Hein. Applying the thorne-kishino-felsenstein model to sequence evolution on a star-shaped tree. *Applied Mathematics Letters*, 14(6):679–684, Aug 2001.

[162] G. D. Stormo and D. S. Fields. Specificity, free energy and information content in protein-dna interactions. *Trends Biochem Sci*, 23(3):109–13, Mar 1998.

[163] J. Stoye, D. Evers, and F. Meyer. Rose: generating sequence families. *Bioinformatics*, 14(2): 157–63, 1998.

[164] C. L. Strope, K. Abel, S. D. Scott, and E. N. Moriyama. Biological sequence simulation for testing complex evolutionary hypotheses: indel-seq-gen version 2.0. *Mol Biol Evol*, 26(11): 2581–93, Nov 2009.

[165] K. Struhl. Gene regulation. a paradigm for precision. *Science*, 293(5532):1054–5, Aug 2001.

[166] A. R. Subramanian, M. Kaufmann, and B. Morgenstern. Dialign-tx: greedy and progressive approaches for segment-based multiple sequence alignment. *Algorithms Mol Biol*, 3:6, 2008.

[167] A. Tanay and E. D. Siggia. Sequence context affects the rate of short insertions and deletions in flies and primates. *Genome Biol*, 9(2):R37, 2008.

[168] A. Tanay, A. Regev, and R. Shamir. Conservation and evolvability in regulatory networks: the evolution of ribosomal regulation in yeast. *Proc Natl Acad Sci U S A*, 102(20):7203–8, May 2005.

[169] J. D. Thompson, D. G. Higgins, and T. J. Gibson. Clustal w: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Res*, 22(22):4673–80, Nov 1994.

[170] J. D. Thompson, F. Plewniak, and O. Poch. A comprehensive comparison of multiple sequence alignment programs. *Nucleic Acids Res*, 27(13):2682–90, Jul 1999.

[171] J. D. Thompson, F. Plewniak, and O. Poch. Balibase: a benchmark alignment database for the evaluation of multiple alignment programs. *Bioinformatics*, 15(1):87–8, Jan 1999.

[172] J. D. Thompson, P. Koehl, R. Ripp, and O. Poch. Balibase 3.0: latest developments of the multiple sequence alignment benchmark. *Proteins*, 61(1):127–36, Oct 2005.

[173] J. L. Thorne, H. Kishino, and J. Felsenstein. An evolutionary model for maximum likelihood alignment of dna sequences. *J. Mol. Evol.*, 33(2):114–124, Aug 1991.

[174] J. L. Thorne, H. Kishino, and J. Felsenstein. Inching toward reality: an improved likelihood model of sequence evolution. *J. Mol. Evol.*, 34(1):3–16, Jan 1992.

[175] D. Tian, Q. Wang, P. Zhang, H. Araki, S. Yang, M. Kreitman, et al. Single-nucleotide mutation rate increases close to insertions/deletions in eukaryotes. *Nature*, 455(7209):105–8, Sep 2008.

[176] I. Van Walle, I. Lasters, and L. Wyns. Sabmark–a benchmark for sequence alignment that covers the entire known fold space. *Bioinformatics*, 21(7):1267–8, Apr 2005.

[177] I. M. Wallace, O. O'Sullivan, and D. G. Higgins. Evaluation of iterative alignment algorithms for multiple alignment. *Bioinformatics*, 21(8):1408–14, Apr 2005.

[178] P. J. Wittkopp. Variable gene expression in eukaryotes: a network perspective. *J Exp Biol*, 210(Pt 9):1567–75, May 2007.

[179] K. M. Wong, M. A. Suchard, and J. P. Huelsenbeck. Alignment uncertainty and genomic analysis. *Science*, 319(5862):473–6, Jan 2008.

[180] W. S. W. Wong and R. Nielsen. Finding cis-regulatory modules in drosophila using phylogenetic hidden markov models. *Bioinformatics*, 23(16):2031–7, Aug 2007.

[181] G. A. Wray. The evolutionary significance of cis-regulatory mutations. *Nat Rev Genet*, 8(3): 206–16, Mar 2007.

[182] X. Xie, J. Lu, E. J. Kulbokas, T. R. Golub, V. Mootha, K. Lindblad-Toh, et al. Systematic discovery of regulatory motifs in human promoters and 3' utrs by comparison of several mammals. *Nature*, 434(7031):338–45, Mar 2005.

[183] Z. Yang. Paml 4: phylogenetic analysis by maximum likelihood. *Mol Biol Evol*, 24(8): 1586–91, Aug 2007.

[184] Z. Yang. Estimating the pattern of nucleotide substitution. *J Mol Evol*, 39(1):105–11, Jul 1994.

[185] Z. Zhang and M. Gerstein. Patterns of nucleotide substitution, insertion and deletion in the human genome inferred from pseudogenes. *Nucleic Acids Res*, 31(18):5338–48, Sep 2003.